AUTOMATIC CONTROL OF SUBMARINE DEPTH, PITCH AND TRIM

Harold Leroy Drurey

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

AUTOMATIC CONTROL OF SUBMARINE DEPTH,
PITCH AND TRIM

by

Harold Leroy Drurey

September 1975

Thesis Advisor: G. J. Thaler

Approved for public release; distribution unlimited.

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Automatic Control of Submarine Depth, Pitch and Trim		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis (September 1975)
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Harold Leroy Drurey		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		12. REPORT DATE September 1975
		13. NUMBER OF PAGES 131
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) Naval Postgraduate School Monterey, California 93940		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Submarine Controls

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A computer program for simulating submarine motion in six degrees of freedom is developed. The simulated submarine is given a capability of shifting ballast. An automatic pitch and depth control is designed for the submarine simulated using optimal control theory. With the depth and pitch in automatic control a trim error signal is developed by comparing the parameters of the plant to that of a linearized model. This error is used to implement an automatic trim control that

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
(Page 1) S/N 0102-014-6601 |

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20.
shifts ballast within the submarine to achieve a neutral trim.

Automatic Control of Submarine Depth, Pitch and Trim

by

Harold Leroy Drurey
Lieutenant, United States Navy
B.S., Naval Postgraduate School, 1974

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
September 1975

ABSTRACT

A computer program for simulating submarine motion in six degrees of freedom is developed. The simulated submarine is given a capability of shifting ballast. An automatic pitch and depth control is designed for the submarine simulated using optimal control theory. With the depth and pitch in automatic control a trim error signal is developed by comparing the parameters of the plant to that of a linearized model. This error is used to implement an automatic trim control that shifts ballast within the submarine to achieve a neutral trim.

TABLE OF CONTENTS

LIST OF FIGURES

I wish to express my sincere appreciation to Dr. George J. Thaler for his guidance and encouragement in the writing of this thesis. My thanks also to Dr. Donald E. Kirk for his assistance in the use of optimal control theory.

I. SUBMARINE CONTROLS

The submarine control problem differs from that of the surface ship in that the submarine translation in three axes and rotation about each of these axes must be considered. A typical submarine implements control about these axes with the use of movable plane control surfaces, shifting of ballast water and the propulsion system.

The submarine operates and is controlled in all six degrees of freedom. To maneuver, three sets of plane surfaces, the propulsion system, and a means of shifting ballast are used. This thesis considered only control of the plane surfaces and the shifting of ballast water. In addition , of the three plane surfaces available only the fairwater planes and stern planes were automatically controlled, i.e. no automatic rudder. The three modes of control were depth, pitch and ballast control. Course control was not considered.

Depth control consists of the sensing of any error between ordered depth and pitch and the actual depth and pitch. The planes were used to implement the control and bring the submarine to the ordered depth and pitch thereby reducing the error to zero or near zero.

Ballast control is the sensing of other than neutral trim of the submarine. Because of surrounding water temperature changes, taking sea water in the ship or pumping it out, the submarine may not weigh the same as the volume of water it displaces. Also too much water forward or aft in the ship initiates a turning moment around the Y axis that will eventually have to be corrected by the planes. This condition must first be detected so a corrective action can be applied by shifting ballast within variable ballast tanks located inside the submarine.

The purpose of controlling depth is obvious. pitch control is required largely because it is coupled to depth control and it is sometimes desired to maintain pitch at other than a zero angle. A means of ballast control is required to surface the ship, maintain depth under certain conditions, allow depth control to be accomplished within the limits of the planes' capabilities and to reduce drag caused by a steady state non zero planes angle. The corrective action for the last two out of trim conditions is normally accomplished by use of a pump to shift water into or out of variable ballast tanks in the submarine.

The method of achieving an automatic control was to first develop an automatic controller that maintained an ordered depth and pitch. The controller was to counteract external forces by movement of the fairwater and stern planes. These forces can be caused by an out of trim condition or by the ship dynamics which at different speeds would cause the submarine to change depth in varying degrees plus such things as sea state, etc. The controller designed is unique to the submarine considered in ref(1) but the method is general and may be applied to most situations. With the depth controller in operation the automatic control of ballast was then solved. Again the solution is unique to the submarine considered. Finally with all automatic controls in operation various runs were made to determine the capability of the system.

However, before a controller design can be started a computer model of the submarine must be developed. Reference 1 contains a FORTRAN program used by NSRDC. A program language that was more easily manipulated in terms of filters, integration routines, etc. was DSL. To use DSL the equations of motion would have to be solved.

II. MODELING A SUBMARINE IN A DIGITAL COMPUTER

A. EQUATIONS OF MOTION FOR A SUBMARINE

The equations of motion in six degrees of freedom are derived by equating the summation of the forces and moments to zero.

Σ Forces=0

Σ Moments=0

Where the vector force

F = d/dt Momentum

=d/dt(mVe) = md/dt Ve

m is the mass and Ve is velocity

Ve = iU+jV+kW

The direction of positive motion is indicated in fig. II-1.

Moment = d/dt(Angular momentum) at the center of gravity.

Moment = d/dt(iPIxx+jQIyy+kRIzz)

Where Ixx is the moment of inertia about the X axis etc. The direction of positive motion is shown in Fig. II-2. A simplified derivation of the equations of motion is found in ref. 2.

The equations are referred to the body, i.e. the origin of the coordinate system is located in the body and at the center of gravity. Figure II-3 is a submarine placed on the coordinate axis with arrows pointing in the direction of positive motion. In addition the movement of the rudder, stern planes and fairwater planes are also depicted on Fig. II-3 with the arrows shown pointing in the direction of positive motion. The specific equations of motion for a submarine were developed for NSRDC and presented in ref. 3. If the coordinate origin is put at the center of gravity

these equations become the the equations of ref. 1. The equations of motion for a submarine in six degrees of freedom are repeated from ref. 1 in the Appendix. For use in the computer programs that follow the equations containing I_x, I_y and I_z had both sides of the equation divided by l^5. Where l is the ship length. In the equations containing m, both sides were divided by $l^3 \rho$. ρ was assigned the value of 2.

The equations contain forcing terms for the planes and rudder, shifting of ballast water and the propeller. The propeller input is a function fitted to a curve. It contains the command speed and actual speed as ratios. Normally there are different sets of coefficients for different propeller modes such as backing, slowing down etc. The only mode considered was forward propulsion at a constant speed. It was noted that the actual speed was always slightly lower than the ordered speed. Uc(command speed) and U(actual speed) are in feet per second.

All angles are in radians. Mass is in slugs and distance in feet.

B. COMPUTER SOLUTION TO THE EQUATIONS OF MOTION

To solve the resultant differential equations ref. 1 partitions the equations into different sums. The equation for U contains no other acceleration term so it is solved separately. V, P and R equations all contain mutual accelerations so they are solved using Cramers Rule for a 3 x 3 system. W and Q contain mutual accelerations so they are grouped and solved as a 2 x 2 system.

The approach chosen in this thesis was to solve the whole system of equations at once using Cramers Rule for a 6 x 6 system. This was more methodical and the determinants were solved only once in the beginning of the program using the "INITIAL" capability of DSL.

The general form of the equations is, using two degrees of freedom for an example

$$S^2 \begin{bmatrix} K_{11} & K_{21} \\ K_{12} & K_{22} \end{bmatrix} \underset{\sim}{X} = \underset{\sim}{F} - \begin{bmatrix} M_{11}S+N_{11} & M_{21}S+N_{21} \\ M_{12}S+N_{12} & M_{22}S+N_{22} \end{bmatrix} \underset{\sim}{X}$$

To simplify the manipulation let

$$\underset{\sim}{I} = \underset{\sim}{F} - \begin{bmatrix} M_{11}S+N_{11} & M_{21}S+N_{21} \\ M_{12}S+N_{12} & M_{22}S+N_{22} \end{bmatrix} \underset{\sim}{X}$$

then by substituting with $\underset{\sim}{I}$

$$S^2 \begin{bmatrix} K_{11} & K_{21} \\ K_{12} & K_{22} \end{bmatrix} \underset{\sim}{X} = \underset{\sim}{I}$$

defining

$$\Delta = \begin{vmatrix} K_{11} & K_{21} \\ K_{12} & K_{22} \end{vmatrix}$$

Using Cramers rule to solve for X_1 and X_2,

$$X_1 = \frac{\begin{vmatrix} I_1 & K_{21}S^2 \\ I_2 & K_{22}S^2 \end{vmatrix}}{S^4 \Delta}$$

$$X_2 = \frac{\begin{vmatrix} K_{11}S^2 & I_1 \\ K_{12}S^2 & I_2 \end{vmatrix}}{S^4 \Delta}$$

Noting the cofactor of $K_{11} = K_{22}$ and the cofactor of $K_{12} = -K_{21}$ etc.

$$X_1 = \frac{I_1 \text{ cofactor } K_{11} + I_2 \text{ cofactor } K_{12}}{S^2 \Delta}$$

$$X_2 = \frac{I_2 \text{ cofactor } K_{22} + I_1 \text{ cofactor } K_{21}}{S^2 \Delta}$$

The same general approach can be used for higher order systems with the number of cofactors taken equal to the number of variables to be solved. In the equations of motion I will be a number of time varying nonlinear functions.

Reference 4 used the library subroutine DTERM in modified form to solve similar equations for the CSMP language. This method was further modified to work with DSL. Six by six determinants were solved by the program to get a solution for the equations in the form of the previous paragraph. By substituting columns and rows of ones and zeros in the row and column where the variables I should be the cofactor was found. For example in a 3 by 3 case

$$\begin{vmatrix} 1 & 0 & 0 \\ 0 & K_{22} & K_{32} \\ 0 & K_{23} & K_{33} \end{vmatrix} = K_{22}K_{33} - K_{32}K_{23} = \text{cofactor } K_{11}$$

$$\begin{vmatrix} K_{11} & 0 & K_{31} \\ K_{12} & 0 & K_{32} \\ 0 & 1 & 0 \end{vmatrix} = K_{12}K_{31} - K_{11}K_{32} = \text{cofactor } K_{23}$$

Program #1 was used to solve for the cofactors in this manner. Once the cofactors were found subsequent programs had the cofactors read in as constants. In the program the cofactors are constants and the variable I is dependent on time and the dynamics of the submarine.

C. SUBMARINE SIMULATION

The computer used for the simulation was the IBM 360/67 located at the W.R. Church Computer Facility, Naval Postgraduate School. The programming language was the Naval Postgraduate School version of DSL/360. Computer program #1 is a simulation of a submarine in motion at 15 knots. The coefficient values used are obtained in ref. 1. As was previously discussed the equations of motion are derived in the body fixed system. A trigonometric conversion of the velocities is used in the auxiliary equations (appendix) to transform the equations to the earth reference system with the origin at x(to), y(to) and z(to).

Reference 1 contains several maneuvers used to check out the equations and coefficients. One maneuver is the vertical plane overshoot where the stern planes are deflected to $+20^{o}$; when the ship is pitched to -5^{o} the stern plane angle is reversed. Program #1 executes the same maneuver with fig. II-4 and II-5 showing depth,pitch and stern plane angle. The results compared favorably with those of ref. 1.

The original equations contained forcing terms that were to represent water being blown from the main ballast tanks. In the programs developed for this thesis the forcing terms represented water being moved in variable ballast tanks with the capability of shifting ballast between tanks or between any tank and sea .

To control the trim of a submarine a means of shifting ballast water must be provided. For the computer simulation a trim pump must also be simulated. Also trim tanks must be positioned in the ship model. The position of the tanks was an arbitrary choice with some traditional basis. The tanks

were positioned as indicated in fig. II-6.

Many options are available in the shifting of variable ballast water. The options depend on the number of tanks and their individual purpose. To simplify the computer simulation the options will be decreased considerably. Only three tanks will be considered. The auxiliary tank is placed at the center of gravity and will be used to adjust any overall mass error. That is a weight differential between the ship and the water it displaces. The forward and after trim tanks will be used only to adjust ballast within the ship. The result will allow negative values for the content of a tank. The options are,

1. Heavy- pump auxiliary tank to sea.
2. Light- flood auxiliary tank from sea.
3. Heavy forward- pump from forward trim tank to after trim tank.
4. Light forward- pump from after trim tank to forward trim tank.

Any ballast correction will be accomplished by first correcting the overall mass error then achieving a forward and aft trim.

Using the pump from Program #2 with Program #1 the effects of shifting ballast was demonstrated in the following simulations. The planes are put at zero and not moved during the run. Figure II-7 is the resultant pitch change when 2000 pounds of ballast is flooded into the auxiliary tank at a rate of 3860 lbs./min. Because of the ship dynamics a 2.7^{o} up angle is taken even though the auxiliary tank is located at the center of gravity. With the planes left at zero the pitch angle remains constant at

this angle. Figure II-8 produces a surprising result, with 2000 pounds flooded in the ship went up 240 feet in 3.5 minutes. (A negative number for depth indicates up and a positive number indicates down.) However keeping in mind that the submarine assumed an up angle the resultant depth change makes more sense. Figure II-9 was plotted with ballast pumped out of auxiliary tank to sea. The ship assumes a $.9^{o}$ degree up angle. The depth is seen in Fig. II-10 to change by 135 feet in 3.5 minutes. Figure II-11 helps to resolve the apparent problem. In fig. II-11 no ballast was transferred yet the ship assumed an up angle of significant degree with the planes on zero. Reference 5 explains that most military submarines will generate a positive pitch due to the ship structure if the planes are zeroed. The submarine in ref. 1 is fictitious and the coefficients were randomly selected but it still appears to have some of the expected characteristics. Finally fig. II-12 and 13 were generated with ballast shifted from after trim to forward trim.

The model performed as expected and the trim pump and planes produced the desired result. The next problem solved was the designing of automatic controls.

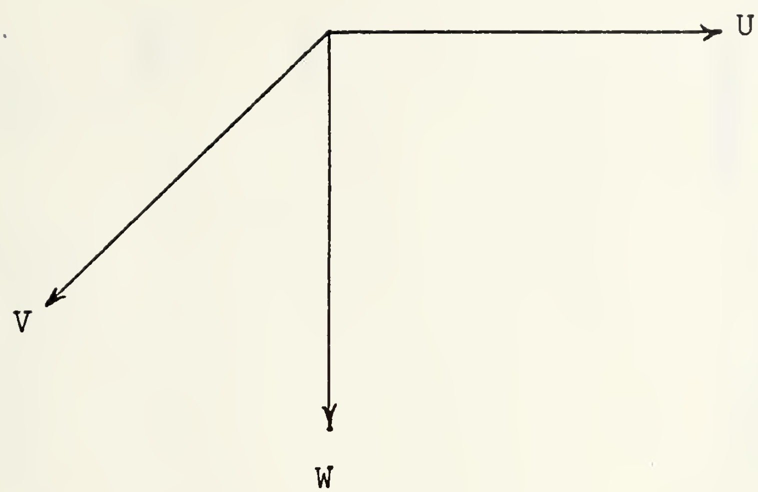


Figure II-1

Axes Definitions

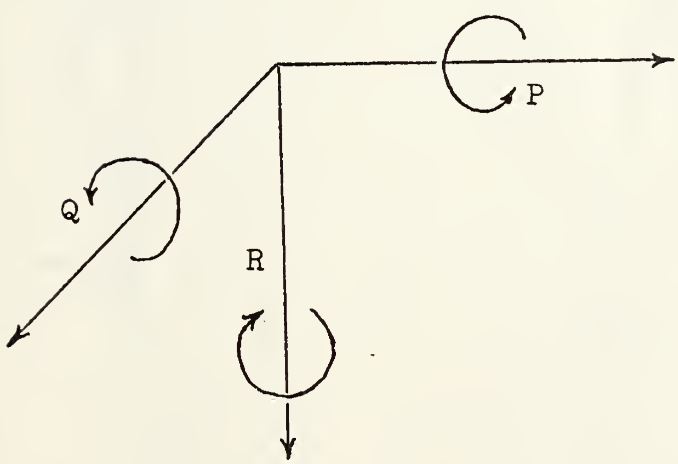


Figure II-2

Axes Definitions

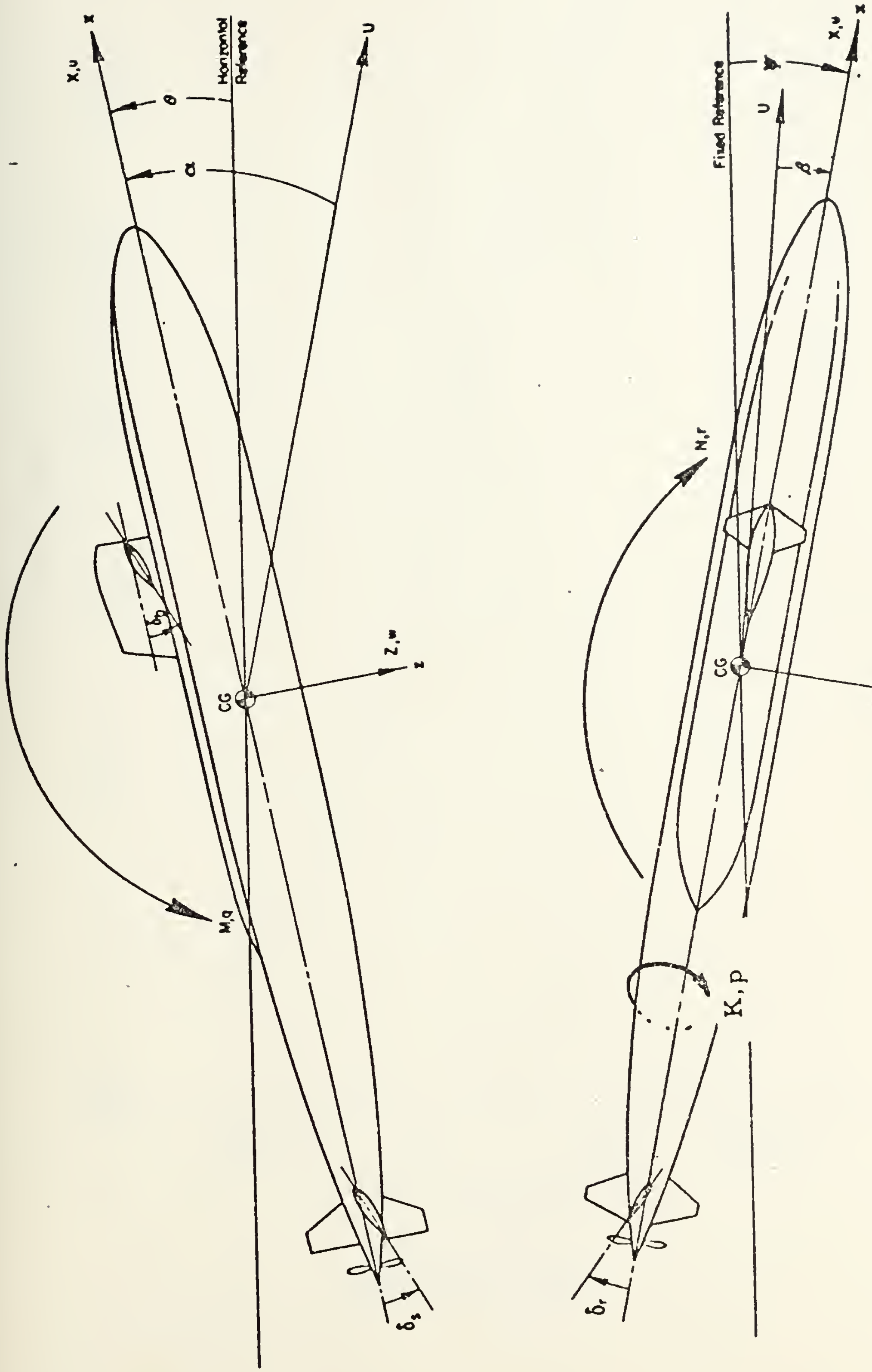


Figure II-3   Submarine Axes

Figure II-4
Depth vs. Time

Figure II-5
Curve 1, Stern Plane Angle vs. Time
Curve 2, pitch vs. Time

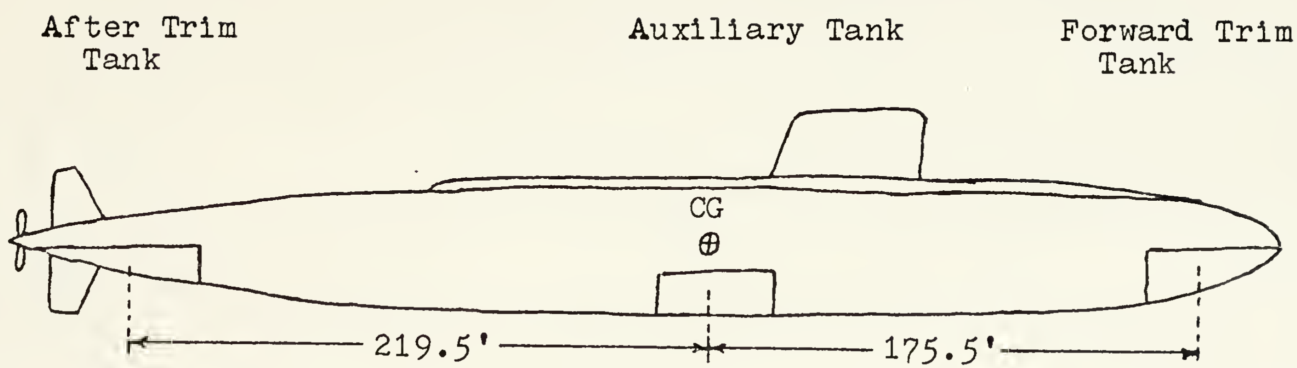


Figure II-6

Tank Locations

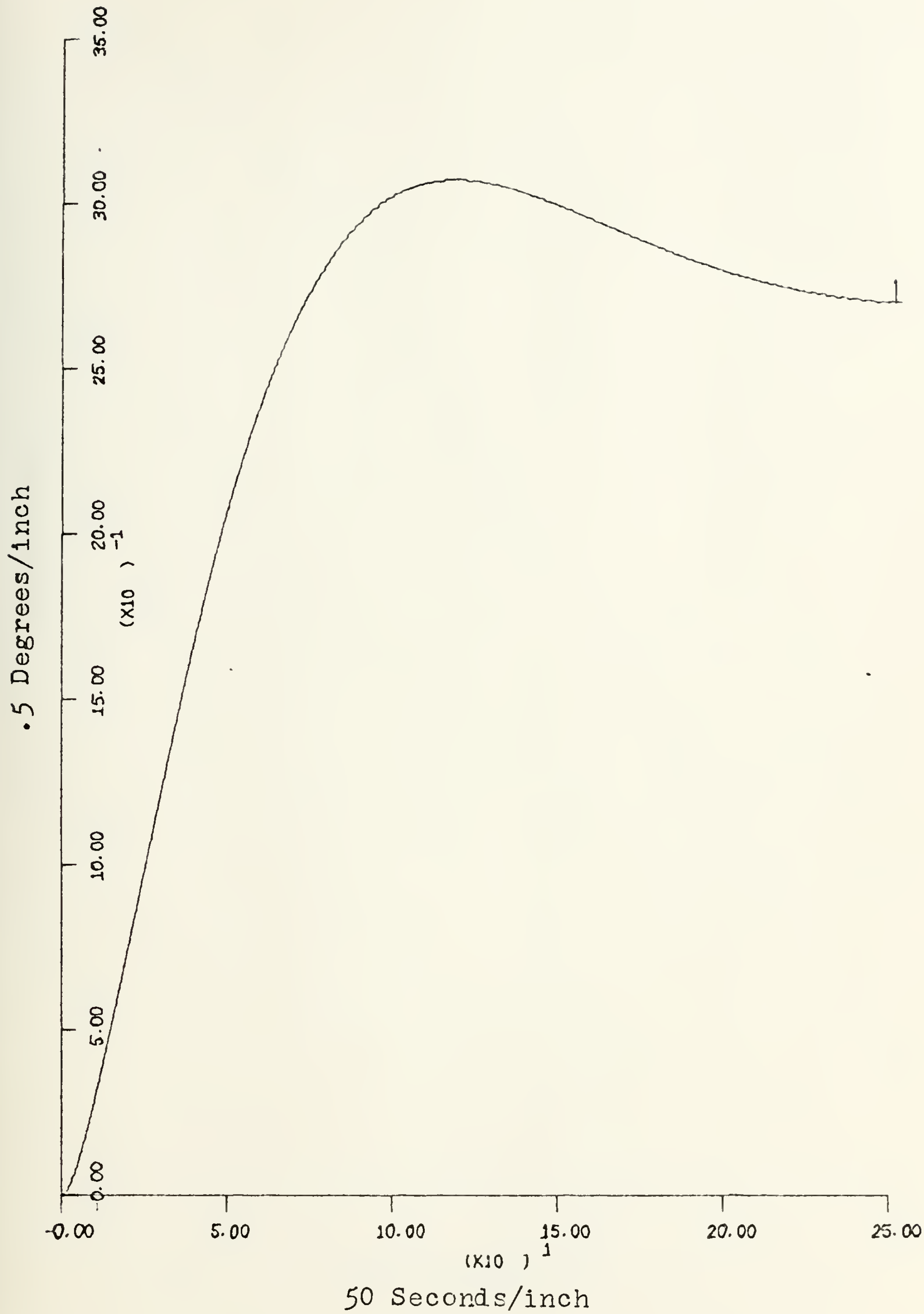


Figure II-7
Pitch vs. Time

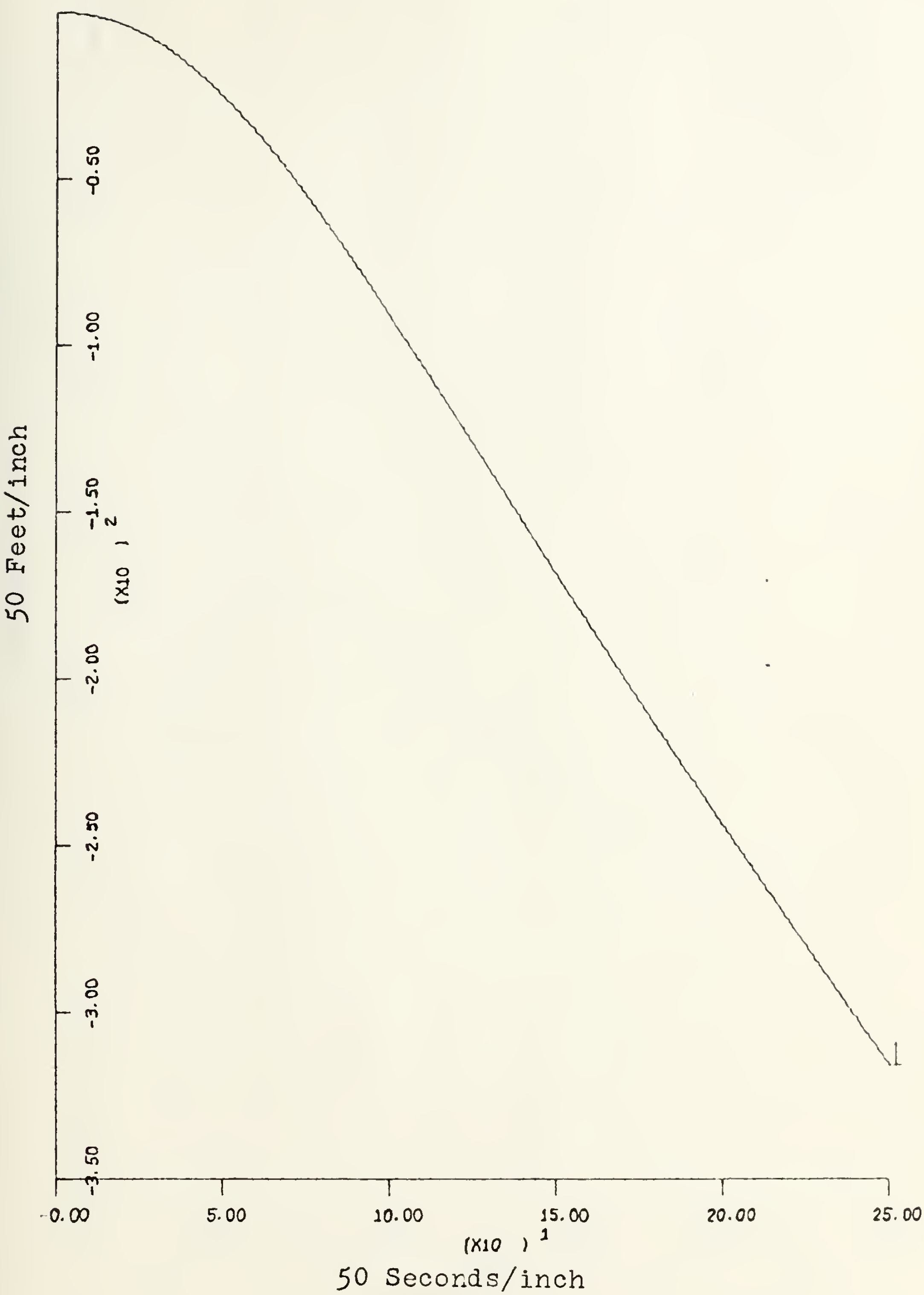


Figure II-8
Depth vs. Time

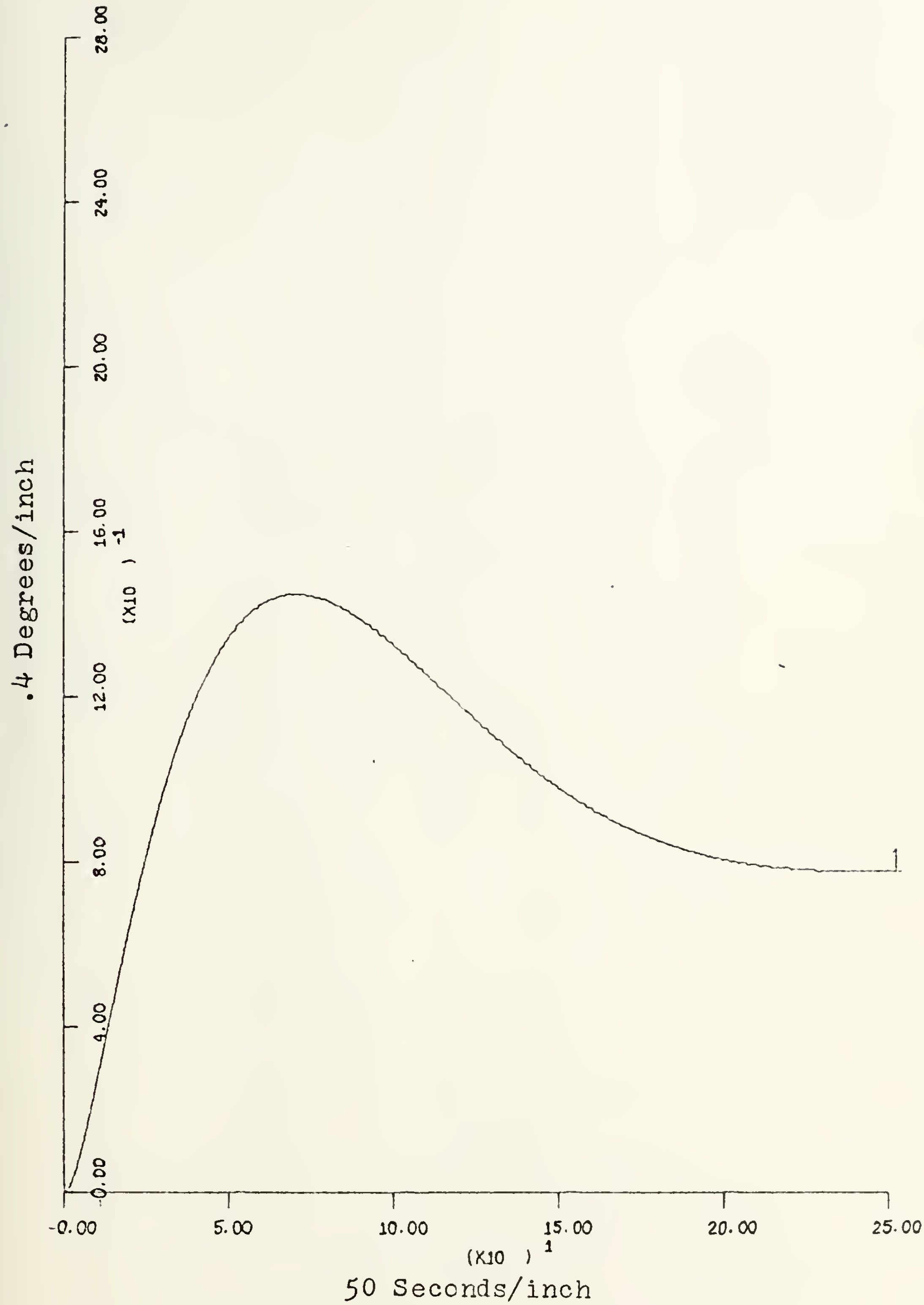


Figure II-9
Pitch vs. Time

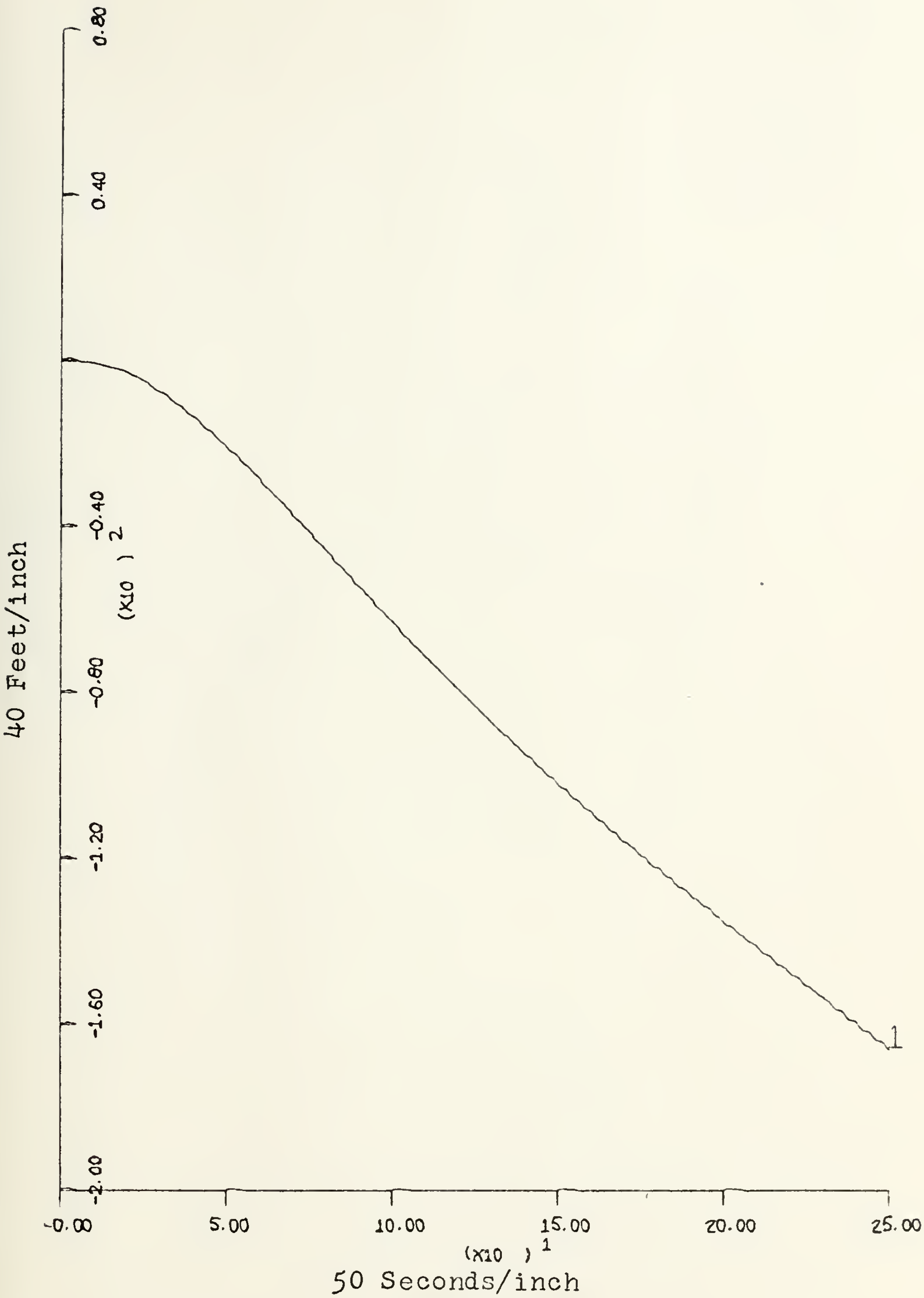


Figure II-10
Depth vs. Time

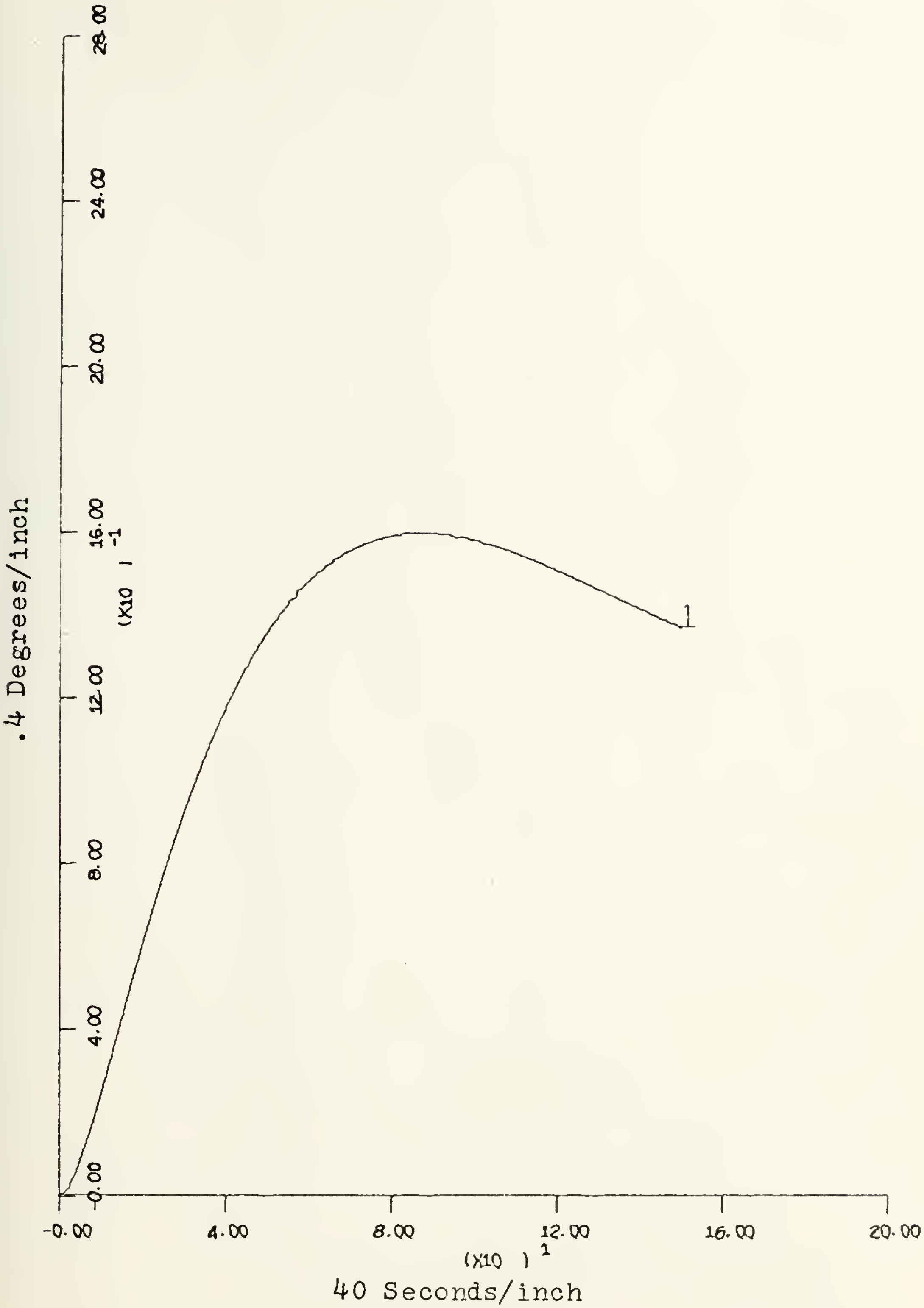


Figure II-11
Pitch vs. Time

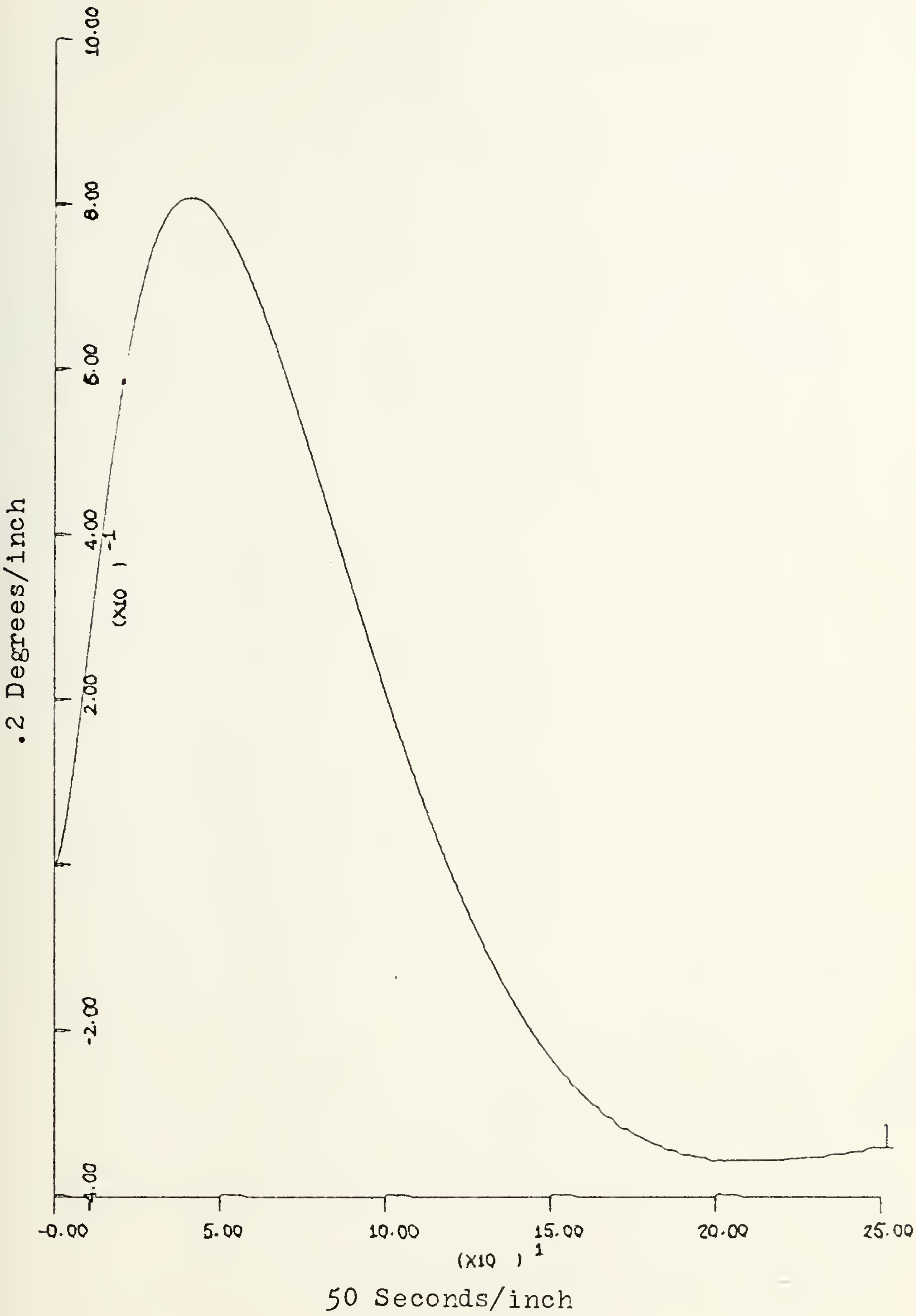


Figure II-12
Pitch vs. Time

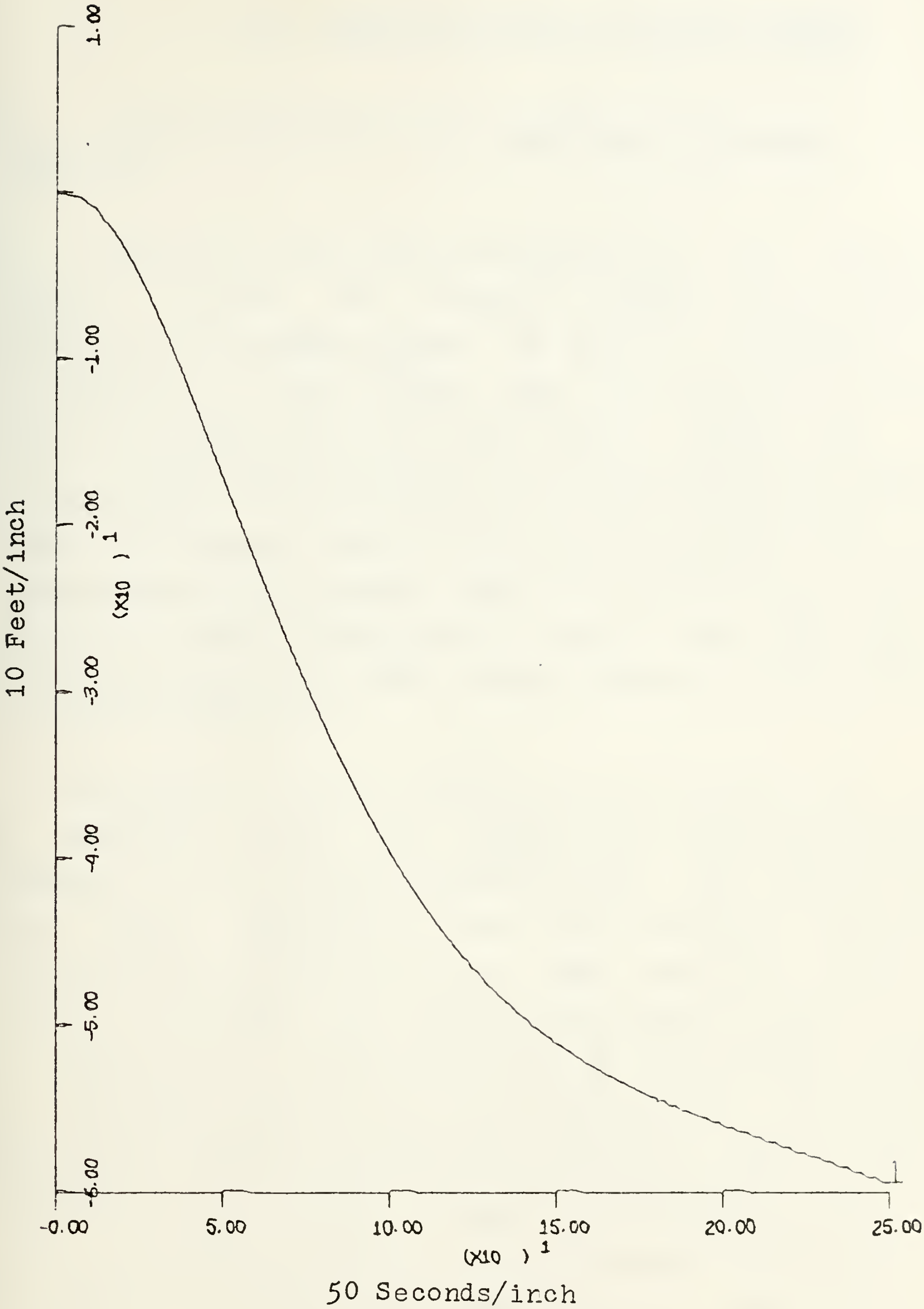


Figure II-13
Depth vs. Time

III. AUTOMATIC PITCH AND DEPTH CONTROL

A. OPTIMAL SOLUTION TO THE LINEARIZED SUBMARINE EQUATIONS

Automatic depth and pitch control can be achieved by various methods. For example, because of the fairwater planes having minimal effect on pitch a frequently used combination is to control the depth with the fairwater planes and pitch with the stern planes. Another way is to set a fixed ratio between stern planes angle and fairwater planes angle. In this case pitch is not "controlled" but is used to achieve ordered depth and the planes are used to change pitch. The approach selected in this thesis was to let the planes individually achieve what ever angle was required to meet an optimal control based on a cost function to be described later.

Because of the nonlinearity of the equations of motion a linearized model was required to solve the equations required for an optimal control system. The required gains were then determined using the linearized equations. Linearizing the equations about some operating point was found to be impractical because the operating point can not be constant and the nonlinearities in the equations present a formidable problem in taking derivatives. Linearization was accomplished by dropping all nonlinear terms with the following justifications. Referring to appendix A all terms involving Wi are dropped because the linearized model is in trim. N'=1 because Uc=U. U is a measured quantity and the only terms of interest are W and Q, i.e. the problem is reduced to two degrees of freedom. The last result is obtained because only the depth and pitch are parameters to be controlled. Simple truncation of the equations of motion will eliminate the other nonlinear terms. The linearized

equations of motion are as follows

$$\begin{bmatrix} m-Zw & -lZq \\ Mw/l & Iy-Mq \end{bmatrix}\begin{bmatrix} \dot{W} \\ \dot{Q} \end{bmatrix} = \begin{bmatrix} UZw/l & UZq \\ UMw/l^{2} & UMq/l \end{bmatrix}\begin{bmatrix} W \\ Q \end{bmatrix}$$

$$+ \begin{bmatrix} U^{2}Zds/l & U^{2}Zdb \\ U^{2}Mds/l^{2} & U^{2}Mdb/l \end{bmatrix}\begin{bmatrix} Ds \\ Db \end{bmatrix}$$

$$\text{let} \quad \begin{bmatrix} m-Zw & lZq \\ Mw/l & Iy-Mq \end{bmatrix}^{-1} = \underset{\sim}{Inv}$$

$$\begin{bmatrix} \dot{W} \\ \dot{Q} \end{bmatrix} = \underset{\sim}{Inv}\begin{bmatrix} UZw/l & UZq \\ UMw/l^{2} & UMq/l \end{bmatrix}\begin{bmatrix} W \\ Q \end{bmatrix} + \underset{\sim}{Inv}\begin{bmatrix} U^{2}Zds/l & U^{2}Zdb/l \\ U^{2}Mds/l^{2} & U^{2}Mdb/l^{2} \end{bmatrix}\begin{bmatrix} Ds \\ Db \end{bmatrix}$$

The values of the inverted matrix are constants and when solved for can be directly substituted as constants. The equations were then written in a more general form that would lend itself to coding for a digital computer program as follows.

$$\begin{bmatrix} \dot{W} \\ \dot{Q} \end{bmatrix} = \begin{bmatrix} A_{11} & A_{21} \\ A_{12} & A_{22} \end{bmatrix}\begin{bmatrix} W \\ Q \end{bmatrix} + \begin{bmatrix} B_{11} & B_{21} \\ B_{12} & B_{22} \end{bmatrix}\begin{bmatrix} Ds \\ Db \end{bmatrix}$$

let

$$Determ=[(Iy-Mq)(m-Zw)-Mw\ Zq]$$

then

$$A_{11}=[(Iy-Mq)Zw+Zq\ Mw]U/l Determ$$

$$A_{12}=[MwZw+(m-Zw)Mw]U/l^{2} Determ$$

$$A_{21}=[(Iy-Mq)Zq+Zq\ Mq]U/Determ$$

$$A_{22}=[MwZq+(m-Zw)Mq]U/l Determ$$

$$B_{11}=[(Iy-Mq)Zds+ZqMds]U^{2}/l Determ$$

$$B_{12}=[MwZds+(m-Zw)Mds]U^{2}/l^{2} Determ$$

$$B_{21}=[(Iy-Mq)Zdb+Zq\ Mdb]U^{2}/l Determ$$

$$B_{22}=[MwZdb+(m-Zw)Mdb]U^{2}/l^{2} Determ$$

The problem was treated as a linear tracking problem with the control designed around the linearized model. The methods of ref. 6 were used for an optimal control with the state equations in the general form

$$\dot{\underset{\sim}{X}} = \underset{\sim}{A}\underset{\sim}{X} + \underset{\sim}{B}\underset{\sim}{U}$$

which represents

$$\begin{bmatrix} \dot{W} \\ \\ \dot{Q} \end{bmatrix} = \underset{\sim}{A} \begin{bmatrix} W \\ \\ Q \end{bmatrix} + \underset{\sim}{B} \begin{bmatrix} Ds \\ \\ Db \end{bmatrix}$$

$$\underset{\sim}{X} \triangleq \begin{bmatrix} W \\ \\ Q \end{bmatrix}$$

and

$$\underset{\sim}{U} \triangleq \begin{bmatrix} Ds \\ \\ Db \end{bmatrix}$$

$\underset{\sim}{A}$ and $\underset{\sim}{B}$ are as previously defined.

The cost function to be minimized is

$$J=1/2 \int_{t_0}^{t_f} [\underset{\sim}{E}^T \underset{\sim}{Q}\underset{\sim}{E} + \underset{\sim}{U}^T \underset{\sim}{R}\underset{\sim}{U}]dt$$

In the derivation of the optimal control equation the control returns the system to the origin. In this controller $\underset{\sim}{r}$ was not restricted to the origin so $\underset{\sim}{X}$ was equivalent to $\underset{\sim}{E}$ where

$$\underset{\sim}{E} = \underset{\sim}{X} - \underset{\sim}{r}$$

and $\underset{\sim}{r}$ is the ordered depth and pitch. $\underset{\sim}{A}, \underset{\sim}{B}, \underset{\sim}{Q}, \underset{\sim}{R}$, and $\underset{\sim}{r}$ are all constant, $\underset{\sim}{X}$ and $\underset{\sim}{U}$ are time varying and the plant is assumed to be completely controllable. (The last assumption was

supported by the solution results.) T_f is considered fixed and $\underset{\sim}{X}(t_f)$ is free. The Hamiltonian is given by

$$\underset{\sim}{Ham} = 1/2[\underset{\sim}{E}^T \underset{\sim\sim}{QE} + \underset{\sim}{U}^T \underset{\sim\sim}{RU}] + \underset{\sim}{p}^T \underset{\sim\sim}{AX} + \underset{\sim}{p}^T \underset{\sim\sim}{BU}$$

The costate equation is

$$\dot{\underset{\sim}{p}}^* = \partial \underset{\sim}{Ham}/\partial \underset{\sim}{X} = -\underset{\sim\sim}{QX}^* - \underset{\sim}{A}^T \underset{\sim}{p}^* + \underset{\sim\sim}{Qr}$$
$$= -\underset{\sim\sim}{QE}^* - \underset{\sim}{A}^T \underset{\sim}{p}^*$$

The asterisks indicate optimal controls and trajectories. The optimal control is then found by

$$\partial Ham/\partial U = \underset{\sim\sim}{RU}^* + \underset{\sim}{B}^T \underset{\sim}{p}^* = 0$$

$$\underset{\sim}{U}^* = -\underset{\sim}{R}^{-1} \underset{\sim}{B}^T \underset{\sim}{p}^*$$

$$\left[\begin{array}{c} \dot{X} \\ \hline \dot{p} \end{array}\right] \quad \left[\begin{array}{c|c} A & -BR^{-1}B^T \\ \hline -Q & -A^T \end{array}\right] \left[\begin{array}{c} E^* \\ \hline p^* \end{array}\right]$$

From ref. 5 the boundary conditions are $\underset{\sim}{p}(t_f) = 0$

defining $\underset{\sim}{p} = \underset{\sim\sim}{KE}$

taking the derivative of $\underset{\sim}{p}$ and dropping the stars

$$\dot{\underset{\sim}{p}} = \underset{\sim\sim}{\dot{K}E} + \underset{\sim\sim}{K\dot{X}}$$

by substitution

$$\dot{p} = \dot{K}E + K\dot{X} = -QE - A^{T}p$$

$$\dot{X} = AE - BR^{-1}B^{T}p$$

$$p = KE$$

then

$$\dot{K} = -KA + KBR^{-1}B^{T}K - Q - A^{T}K$$

and

$$U = -R^{-1}B^{T}p = -R^{-1}B^{T}KE$$

As described in ref. 5 $K(t_f) = 0.0$. K is symmetric i.e. $K_{12}=K_{21}$ etc. so $n(n+1)/2$ differential equations must be solved where n=4. The solution of the differential equation for K will provide the optimal gains for U.

Referring to the original equations and putting them in state variable form

$$
\begin{bmatrix} \dot{X}_1 \\ \dot{X}_2 \\ \dot{X}_3 \\ \dot{X}_4 \end{bmatrix}
=
\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & A_{11} & 0 & A_{21} \\ 0 & 0 & 0 & 1 \\ 0 & A_{12} & 0 & A_{22} \end{bmatrix}
\begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix}
+
\begin{bmatrix} 0 & 0 \\ B_{11} & B_{21} \\ 0 & 0 \\ B_{12} & B_{22} \end{bmatrix}
\begin{bmatrix} Ds \\ Db \end{bmatrix}
$$

$$
\underset{\sim}{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & A_{11} & 0 & A_{21} \\ 0 & 0 & 0 & 1 \\ 0 & A_{12} & 0 & A_{22} \end{bmatrix}
$$

$$
\underset{\sim}{B} = \begin{bmatrix} 0 & 0 \\ B_{11} & B_{21} \\ 0 & 0 \\ B_{12} & B_{22} \end{bmatrix}
$$

$$
\underset{\sim}{Q} = \begin{bmatrix} D & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & E & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}
$$

$$
\underset{\sim}{R} = \begin{bmatrix} C & 0 \\ 0 & C \end{bmatrix}
$$

$\underset{\sim}{R}$ and $\underset{\sim}{Q}$ are the weighting matrices.

The following equations result from the matrix manipulation.

$$F_{11} = B_{11}K_{12} + B_{12}K_{14}$$

$$F_{21} = B_{11}K_{22} + B_{12}K_{24}$$

$$F_{31} = B_{11}K_{23} + B_{12}K_{34}$$

$$F_{41} = B_{11}K_{24} + B_{12}K_{44}$$

$$F_{12} = B_{21}K_{12} + B_{22}K_{14}$$

$$F_{22} = B_{21}K_{22} + B_{22}K_{24}$$

$$F^{32}_{\cancel{23}} = B_{21}K_{23} + B_{22}K_{34}$$

$$F^{42}_{\cancel{24}} = B_{21}K_{24} + B_{22}K_{44}$$

$$\dot{K}_{11} = -E + (F_{11}^2 + F_{12}^2)/C$$

$$\dot{K}_{12} = -(K_{11} + K_{12}A_{11} + K_{14}A_{12}) + (F_{21}F_{11} + F_{22}F_{12})/C$$

$$\dot{K}_{13} = (F_{31}F_{11} + F_{32}F_{12})/C$$

$$\dot{K}_{14} = -(K_{12}A_{12} + K_{13} + K_{14}A_{22}) + (F_{41}F_{11} + F_{42}F_{12})/C$$

$$\dot{K}_{22} = -2(K_{12} + K_{22}A_{11} + K_{24}A_{12}) + (F_{21}^2 + F_{22}^2)/C$$

$$\dot{K}_{23} = -(K_{13} + K_{23}A_{11} + K_{34}A_{12}) + (F_{31}F_{21} + F_{32}F_{22}/C$$

$$\dot{K}_{24} = -(K_{14} + K_{24}A_{11} + K_{44}A_{12}) - (K_{22}A_{21} + K_{23} + K_{24}A_{22}) +$$
$$(F_{41}F_{21} + F_{42}F_{22})/C$$

$$\dot{K}_{33} = (F_{31}^2 + F_{32}^2)/C - D$$

$$\dot{K}_{34} = -(K_{23}A_{21} + K_{33} + K_{34}A_{22}) + (F_{41}F_{31} + F_{42}F_{32})/C$$

$$\dot{K}_{44} = -2(K_{24}A_{21} + K_{34} + K_{44}A_{22}) + (F_{41}^2 + F_{42}^2)/C$$

Since only the value of $\underset{\sim}{K}(t_f)$ is known these equations must

be solved backwards i.e. from t_f to t_o. To do this let

$$\tau = t_f - t$$

$$d\tau = -dt$$

when $t=t_f$, $\tau=0$

$$\dot{\underset{\sim}{K}} = d\underset{\sim}{K}/dt = -d\underset{\sim}{K}/d\tau$$

The signs of all the equations are reversed and the initial conditions are those conditions at t_f. Up to this point the solution is general and when solved the solution will provide gains for

$$\underset{\sim}{U} = -\underset{\sim}{R}^{-1}\underset{\sim}{B}^{T}\underset{\sim}{K}\underset{\sim}{E}$$

$$= -1/C \begin{bmatrix} F_{11} & F_{21} & F_{31} & F_{41} \\ F_{12} & F_{22} & F_{32} & F_{42} \end{bmatrix} \underset{\sim}{E}$$

The calculations of the gains that follow apply only to the submarine in ref. 1. The submarine described in ref. 1 is fictitious and used for demonstration purposes only. DSL again is used to solve for the gains. The program used is program #2 which also contains the controllers designed in the following sections.

Program #2 first calculates the steady state gains used in the controller. While the gains are calculated the ship simulation is held at zero. After a specified time the simulation starts with the gains determined by the specified weighting matrix.

Selection of the weighting matrix to be used can be

started off with some judgment as to the severity of different errors. For instance the pitch equations are in radians and the depth equations are in feet. It is obvious that a one radian pitch error is more severe than a one foot depth error. This immediately would lead to a higher weight on the pitch error signal. It was found that if too much emphasis was placed on pitch the depth error in steady state would become unacceptable. This lead to a trial and error process to select an acceptable weighting matrix combination that would give the desired results.

To operate as an optimal controller the gains would have to be generated on line. This would require a significant computer capability. In an infinite duration process the gains would be constant as demonstrated in fig. III-1 through 7. The optimal control can then be approximated with a steady state gain if the control interval is long with respect to the decay time of the optimal gains. Figure III-1 through III-7 indicate a relatively short rise and fall time for the gains just before t_f.

As determined by observing the equations for K the gains are functions of the velocity U. A number of different runs were made at different velocities and the following approximate proportionalities were observed. K_{12}, K_{14}, K_{23} and K_{34} are inversely proportional to U^2. K_{22}, K_{24}, and K_{44} are inversely proportional to U^3. The runs were made at 25.33 feet per second so the following adjustment was made to the gains used in the programs.

$$ZM1=(25.33)^2 \; K_{12}/U^2$$

$$ZM2=(25.33)^2 \; K_{14}/U^2$$

$$ZM3=(25.33)^3 \; K_{22}/U^3$$

$$ZM4=(25.33)^2 \; K_{23}/1^2$$

$$ZM5=(25.33)^3 \; K_{24}/U^3$$

$$ZM6=(25.33)^2 \; K_{34}/U^2$$

$$ZM7=(25.33)^3 \; K_{44}/U^3$$

The new gains are given by ZM1 through ZM7. The B matrix contains U^2 so a cancellation of U takes place in the final equation.

B. AUTOMATIC PITCH AND DEPTH CONTROL

Figure III-8 and III-9 were simulations with C=1.0, D=100.0 and E=1.0 or the control input was weighted to 1.0, the pitch error was weighted to 100.0, and depth error weighted to 1.0. The ordered depth change was 10.0 feet. The controller initiated the proper planes but because the pitch was so large when ordered depth was achieved the overshoot pushed the control system into instability. Also the ship pitches in the wrong direction. The initial plane response is to a depth error and both planes were positioned as seen in fig. III-10. The ship pitched up with an angle too large to overcome.

The next run was made after weighting the pitch and the

control input more heavily. The weighting matrix used was C=10.0, D=300.0, and E=1.0. Figure III-11 is the resultant depth change for a 10 foot step input in ordered depth. The resultant steady state depth error was .29 feet which was acceptable. As before the ship initially pitched up to about 12^{o} and then returned to a steady state angle of $+1.28^{o}$ as seen in fig. III-12. Figure III-13 shows the fairwater and stern plane angles used to achieve the new ordered depth. The maximum -80^{o} stern plane angle and -50^{o} fairwater plane angle are unachievable on a real ship and the rates of change of depth and pitch were unnacceptable. When a limit constraint was placed on the plane angles the system went into a limit cycle. The steady state error in pitch was large enough to prevent the trim controller in the next section from detecting a ballast change to any degree of accuracy. Lastly the pitch variation of 12^{o} was considered unacceptable for only a 10 foot depth change.

Larger emphasis was required on pitch but this caused the plane angles to become larger and if the weighting on plane angles was increased the system became insensitive to either depth error or pitch error depending on the weight. The problem appeared to be a result of the initial depth error. To limit this effect with a large step input the depth and pitch error were limited as shown in fig. III-14. Also a larger emphasis was required on pitch error in the weighting matrix. The weighting matrix chosen was C=10.0, D=3000.0, and E=1.0.

In fig. III-13 a significant noise level is seen in the plane positions. To reduce this noise the plane angle ordered was filtered with a simple low pass filter.

The tests were run again and this time a 20 foot depth

change was ordered. The magnitude of the saturation limit can be adjusted for different results . The values selected for saturation were ± two feet for depth error and ± 10° for pitch error. Major differences to be noted in the runs are the reduced rate of depth change. A 20 foot depth change took about 45 seconds. The maximum pitch assumed was -.8° . The steady state depth error was .2 feet and the steady state pitch error was .14° . This run was plotted in fig III-15, 16 and 17.

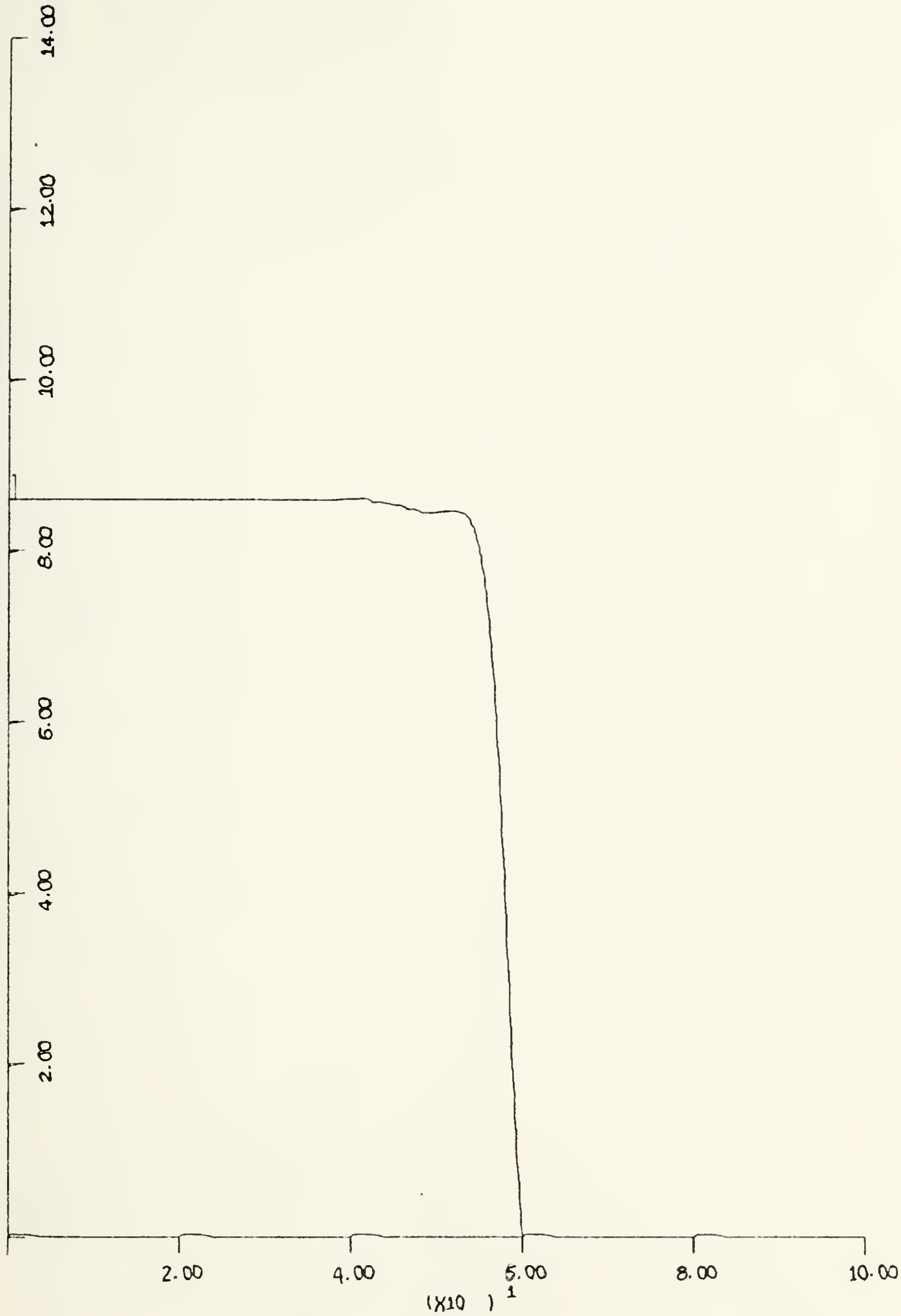
The controller was tested for its ability to maintain depth. To check this the ordered depth was put at zero. The controller maintained a steady state depth error of .16 feet and a steady state pitch error of .14° as seen in fig. III-18 and III-19. The plane angles are presented in fig. III-20. Note that a steady state fairwater plane angle of -4.0° was required to maintain depth. What this implies is that to run the simulated submarine at a zero pitch and on the ordered depth with zero plane angles, ballast would have to be added.

The depth controller was also required to maintain depth in a turn. A simulated rudder angle of 10° was added. The rudder motion is not restrained in rate for this simulation. After 50 seconds the rudder angle was put to zero. Figure III-21 plots the roll developed as a result of the manipulation. The transient in pitch and depth was small as indicated in fig. III-22 and 23. In particular fig. III-24 shows a positive fairwater and stern plane angle required as the submarine settles in the stern.

In early runs it was apparent that the submarine was "light" and ballast should be added. This was based on a

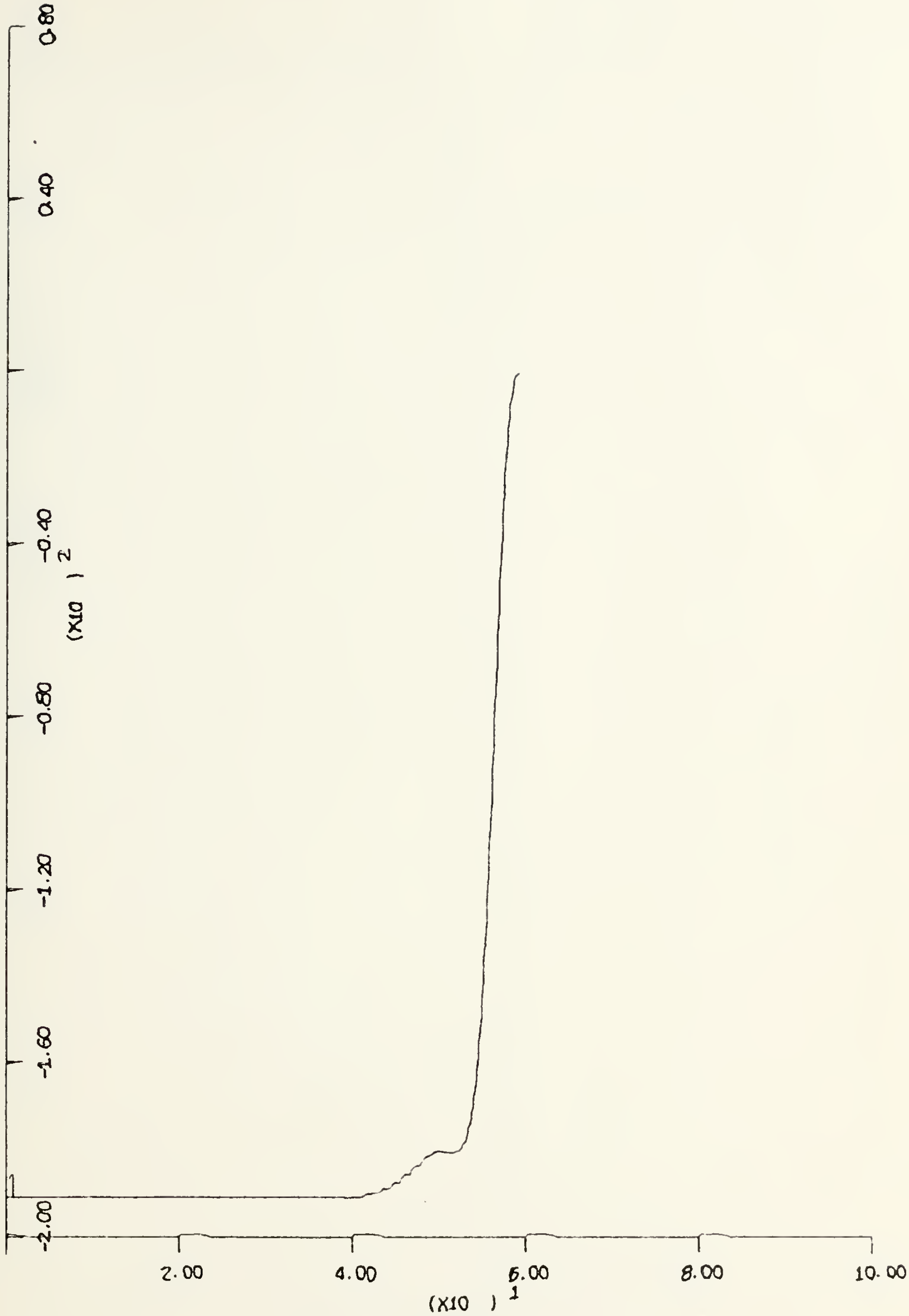
steady state fairwater plane angle of about -4.0° with a near zero depth and pitch error. In the following run ballast was added to the auxiliary tank at about 8000 pounds per minute for 2 minutes while the planes were in automatic control. Figure III-25 and 26 show a steady state depth error and pitch error of about -.02 feet and $.04^{\circ}$ as compared to .16 feet and $.14^{\circ}$ with out the ballast. Also the steady state fairwater plane angle, Fig. III-27 , shows a steady state angle of about -1.5° compared to -4.0°. This confirms that ballast should be added to run the ship in "trim".

The depth controller was operating satisfactorily with a controller that is somewhat less than optimal for large transients and near optimal for small transients. The last problem to solve was to determine ballast error in the submarine and correct it.



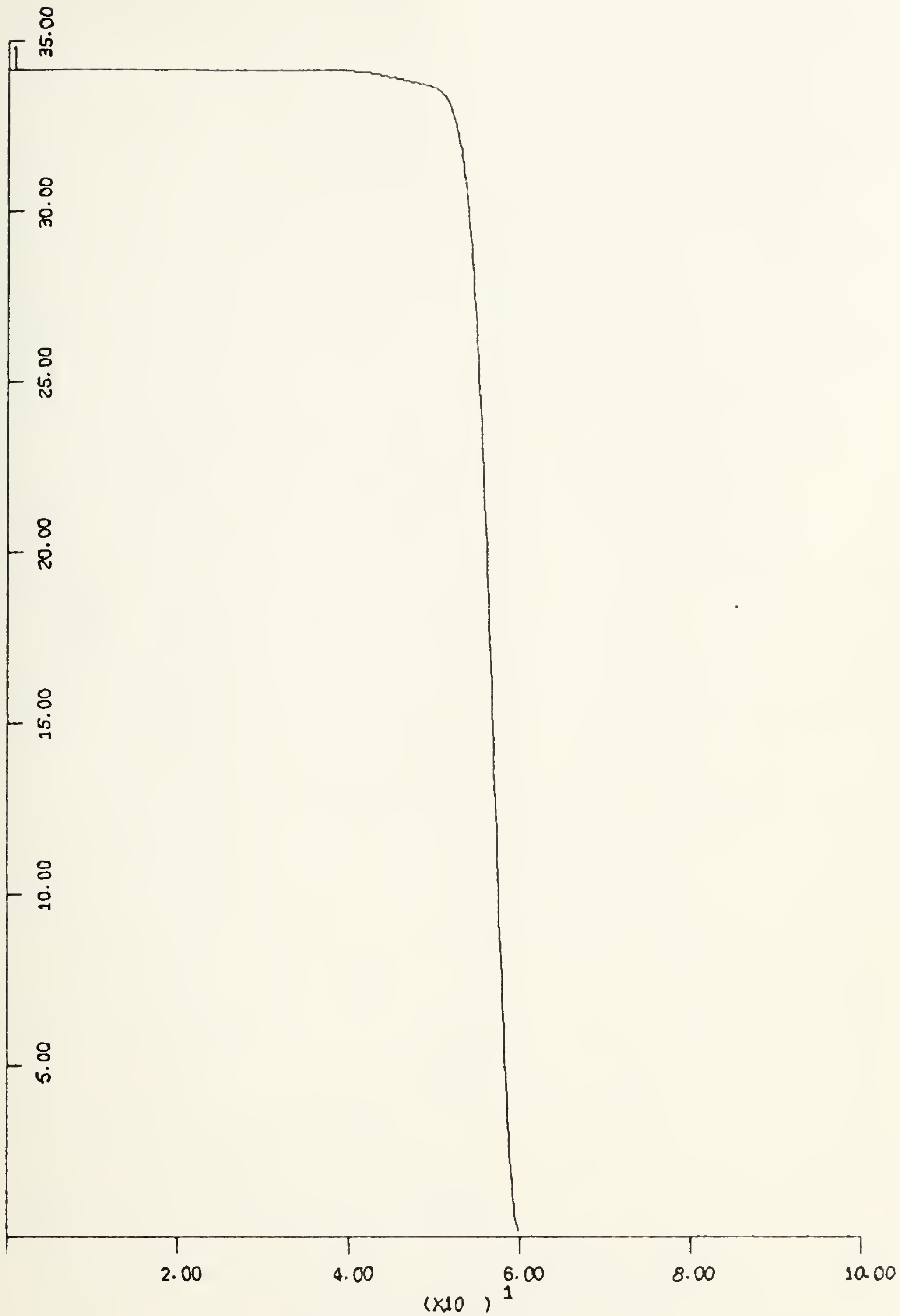
20 Seconds/inch

Figure III-1
K12 vs. Time



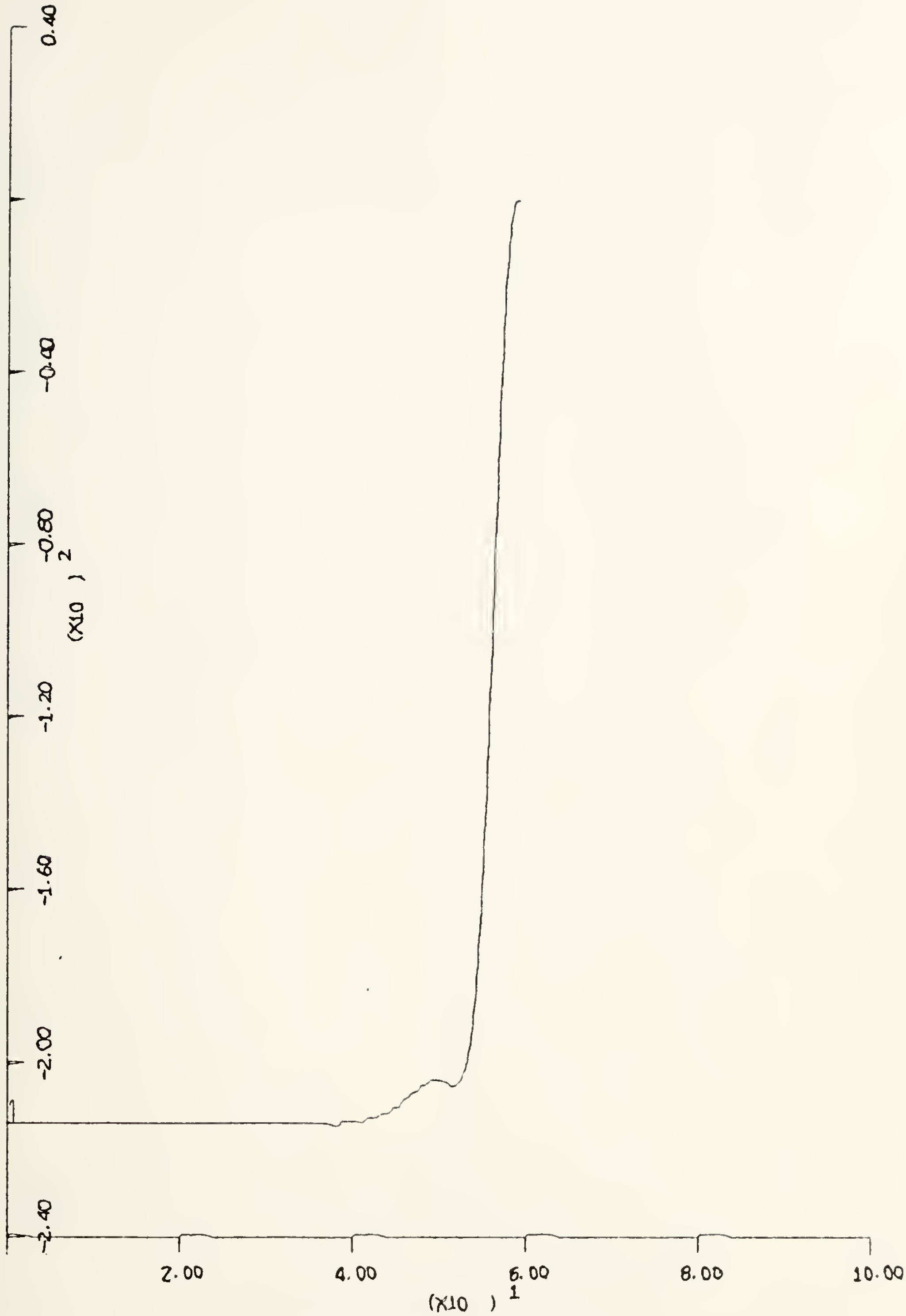
20 Seconds/inch

Figure III-2
K14 vs. Time



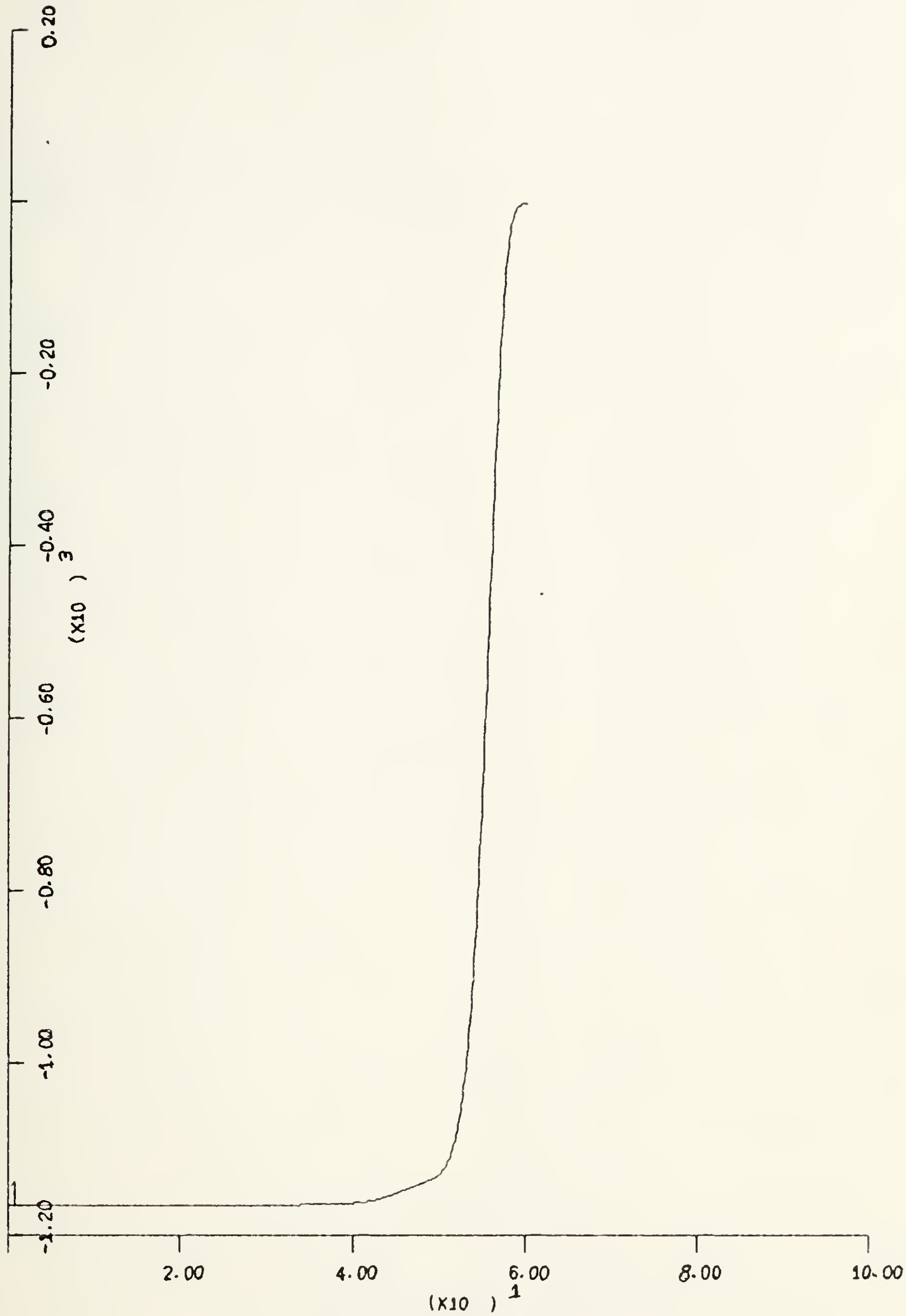
20 Seconds/inch

Figure III-3
K22 vs. Time



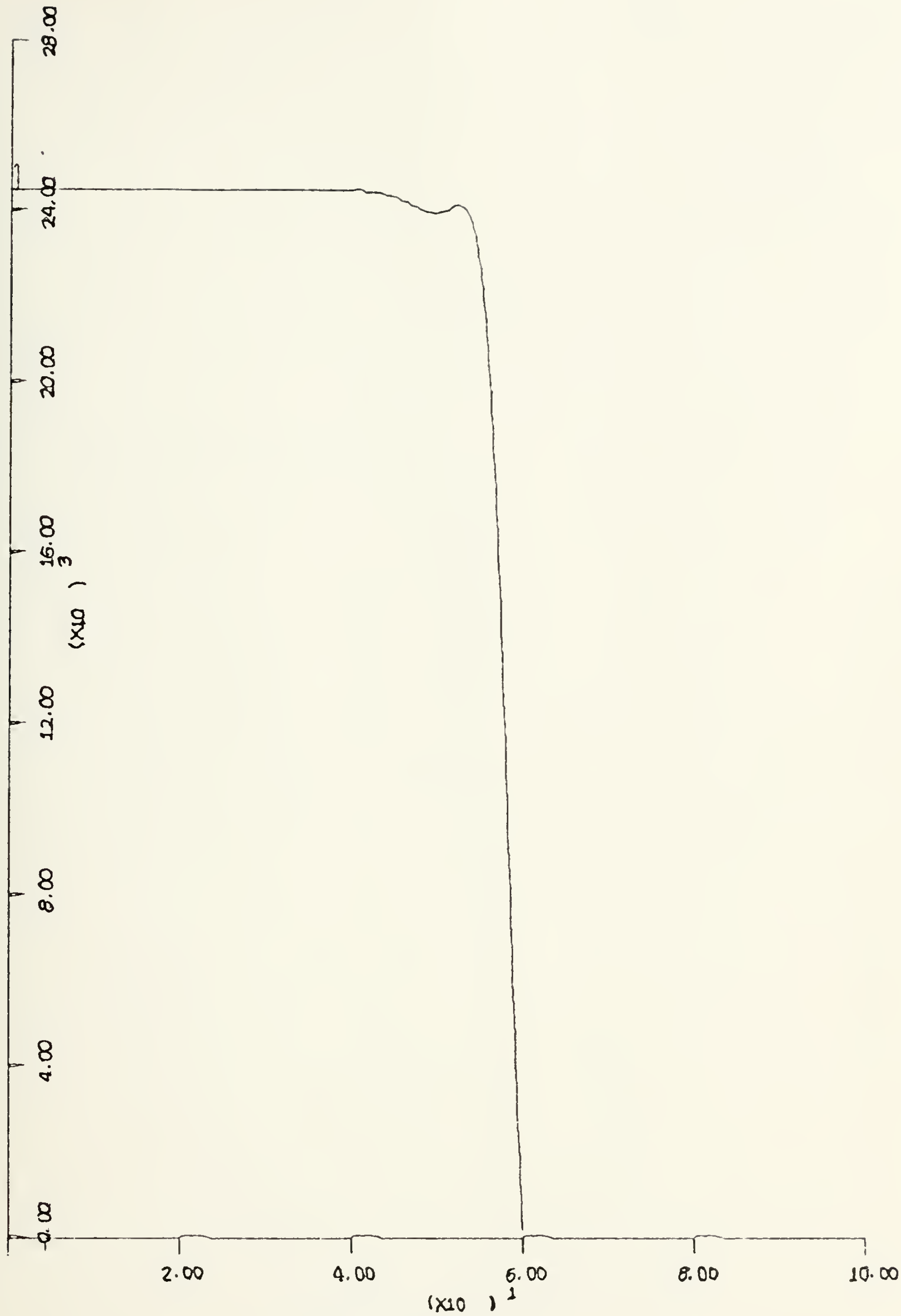
20 Seconds/inch

Figure III-4
K23 vs. Time



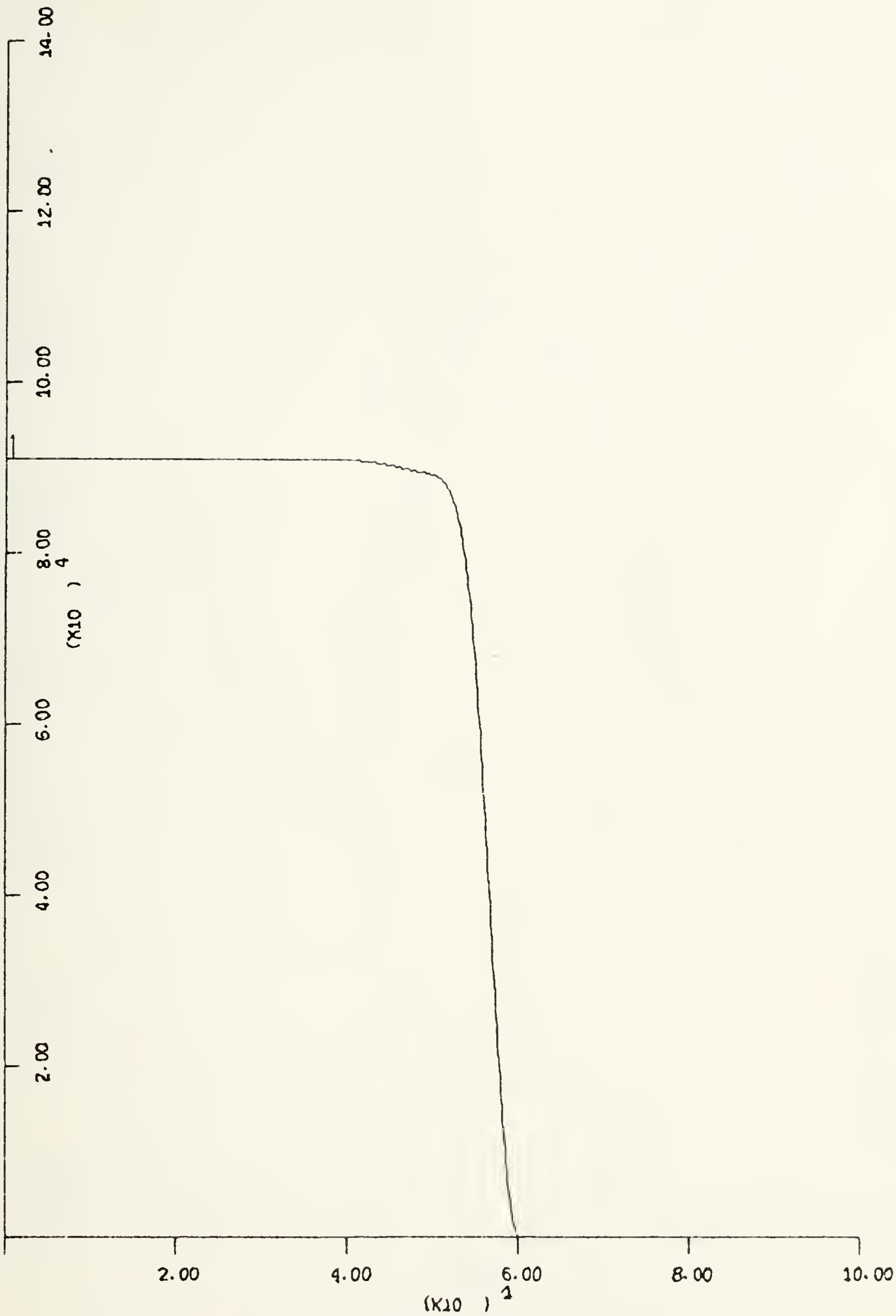
20 Seconds/inch

Figure III-5
K24 vs. Time



20 Seconds/inch

Figure III-6
K34 vs. Time



20 Seconds/inch

Figure III-7
K44 vs. Time

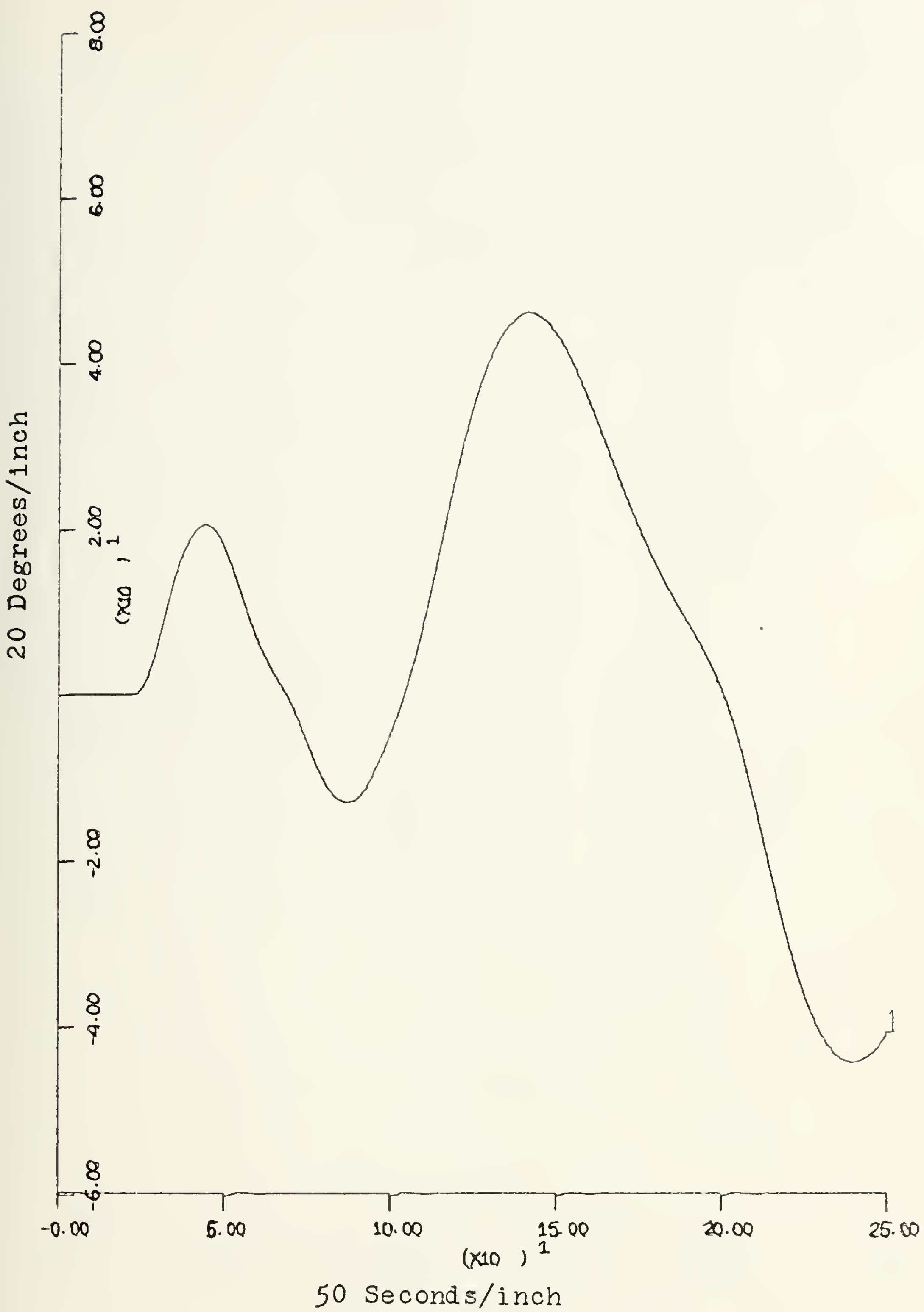


Figure III-8
Pitch vs. Time

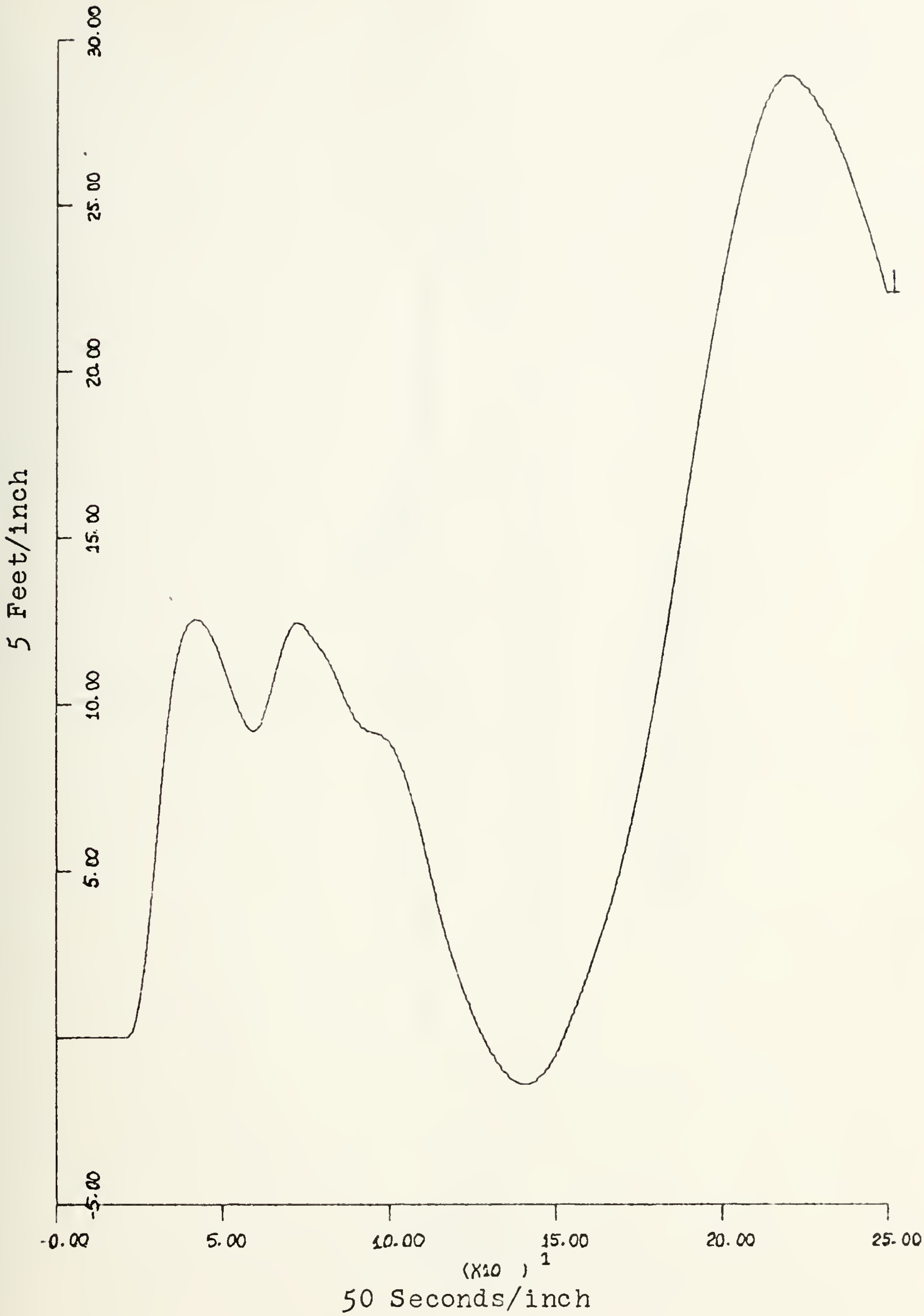


Figure III-9
Depth vs. Time

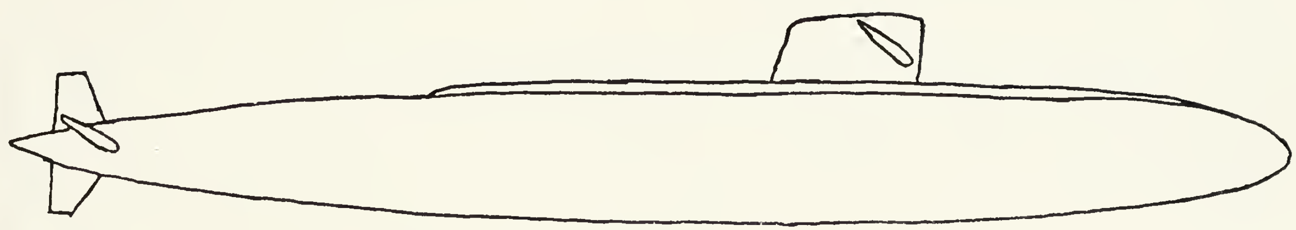
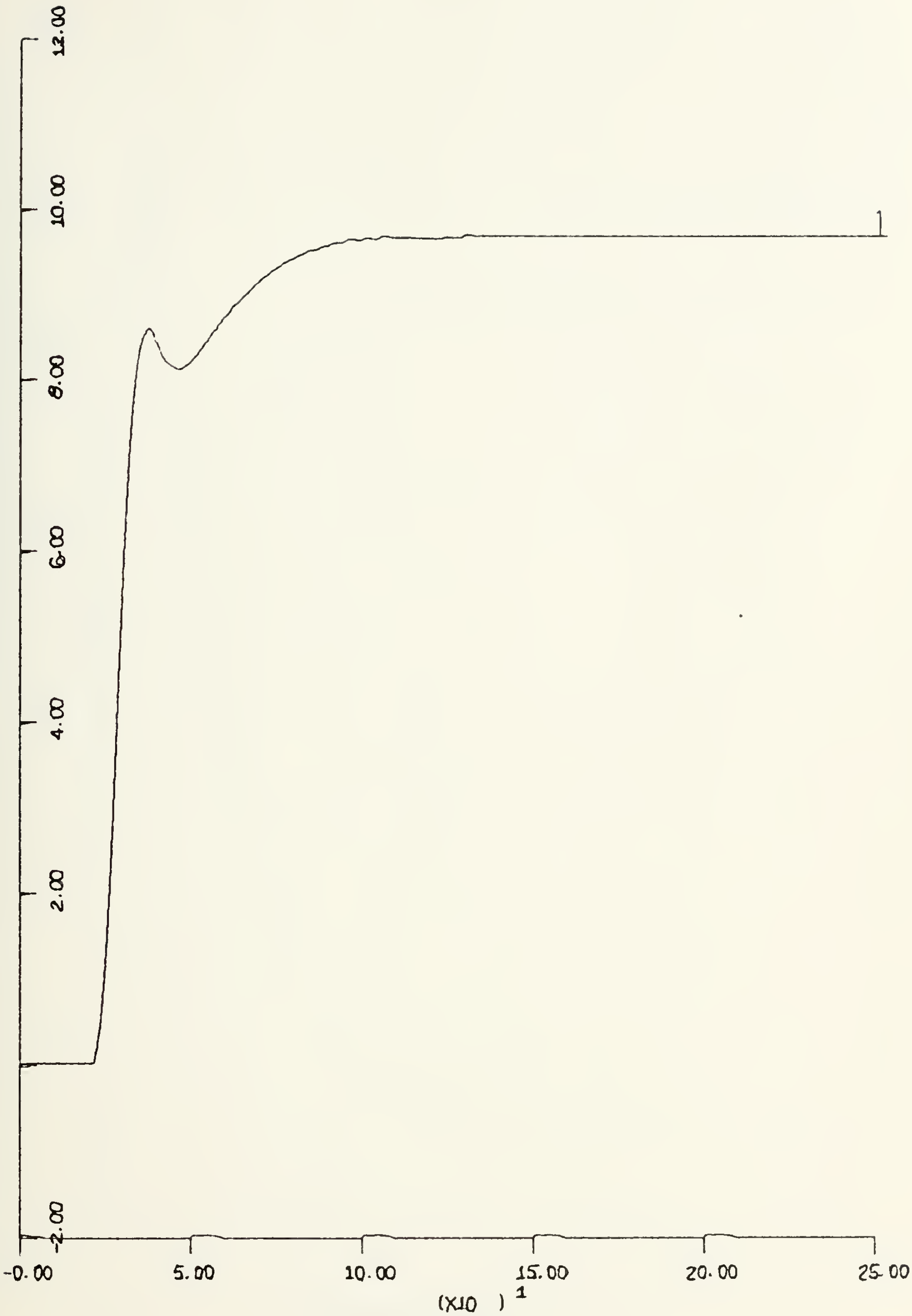


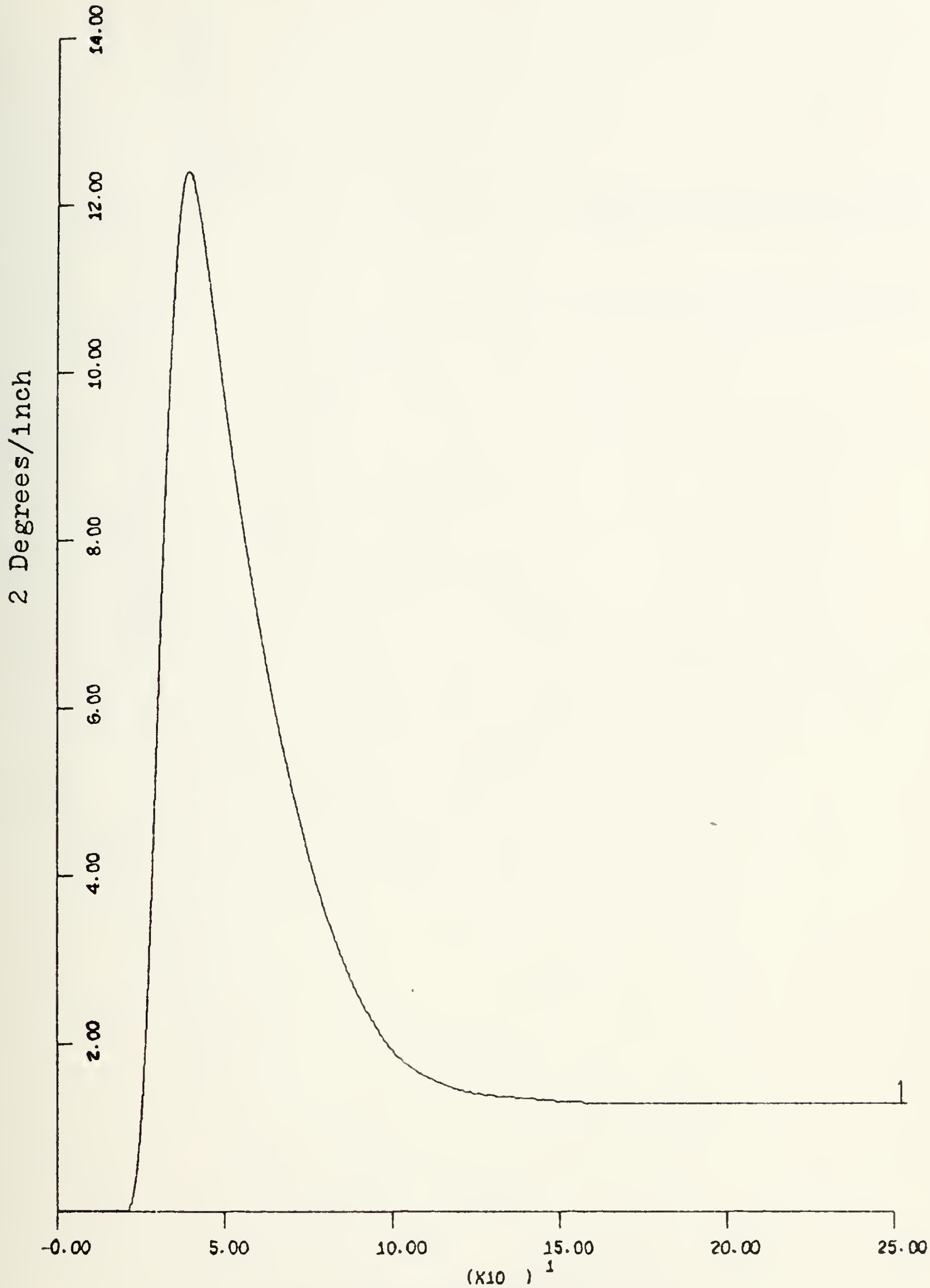
Figure III-10
Plane Angles Response

2 Feet/inch



50 Seconds/inch

Figure III-11
Depth vs. Time



50 Seconds/inch

Figure III-12
Pitch vs. Time

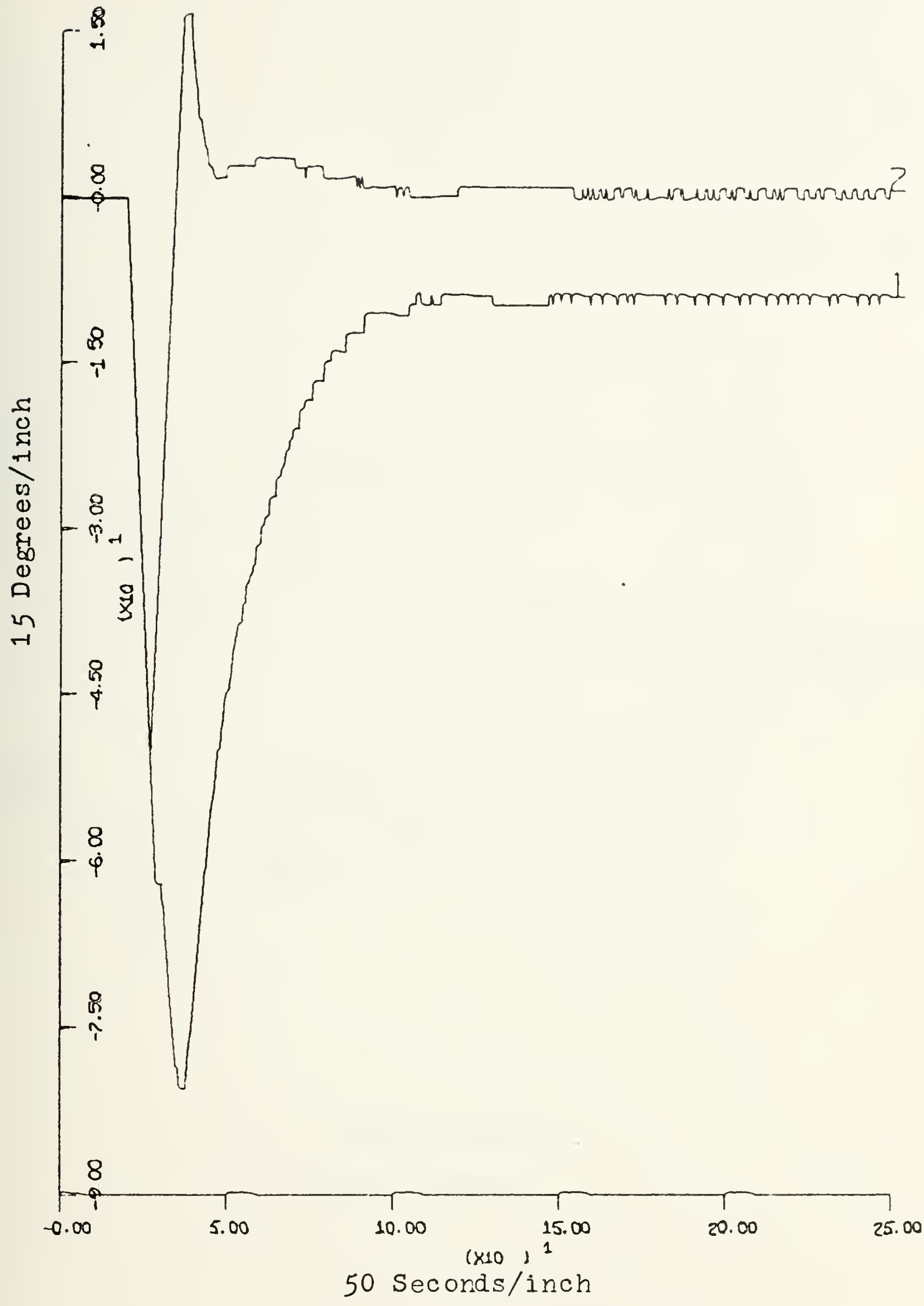


Figure III-13
Curve 1, Stern Planes Angle vs. Time
Curve 2, Fairwater Planes Angle vs. Time

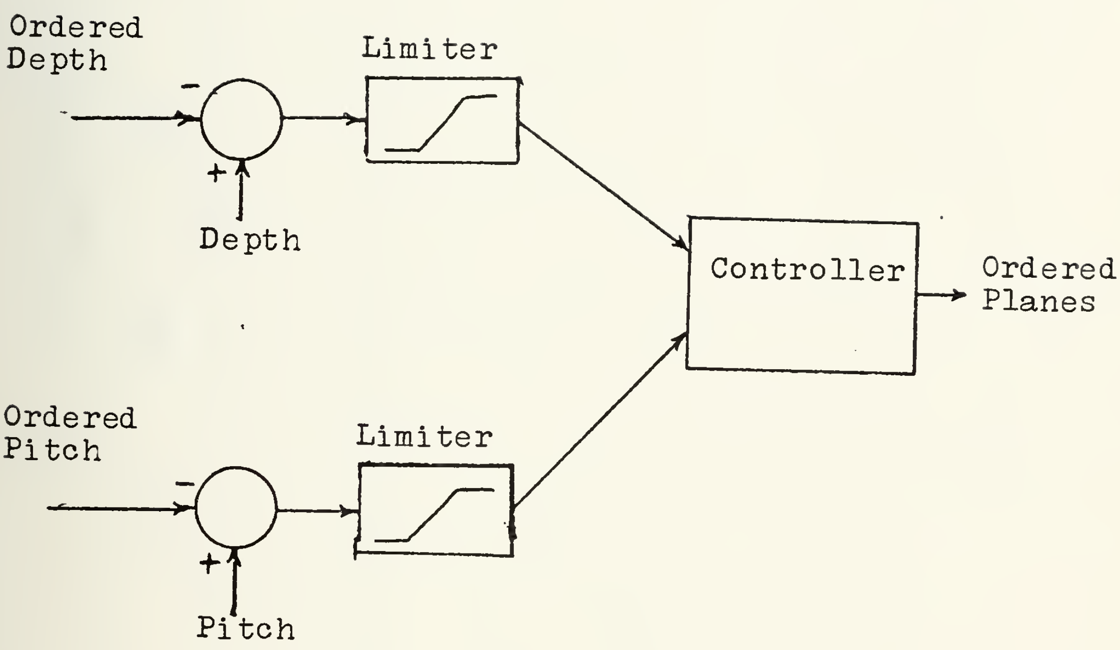


Figure III-14

Error Limiter

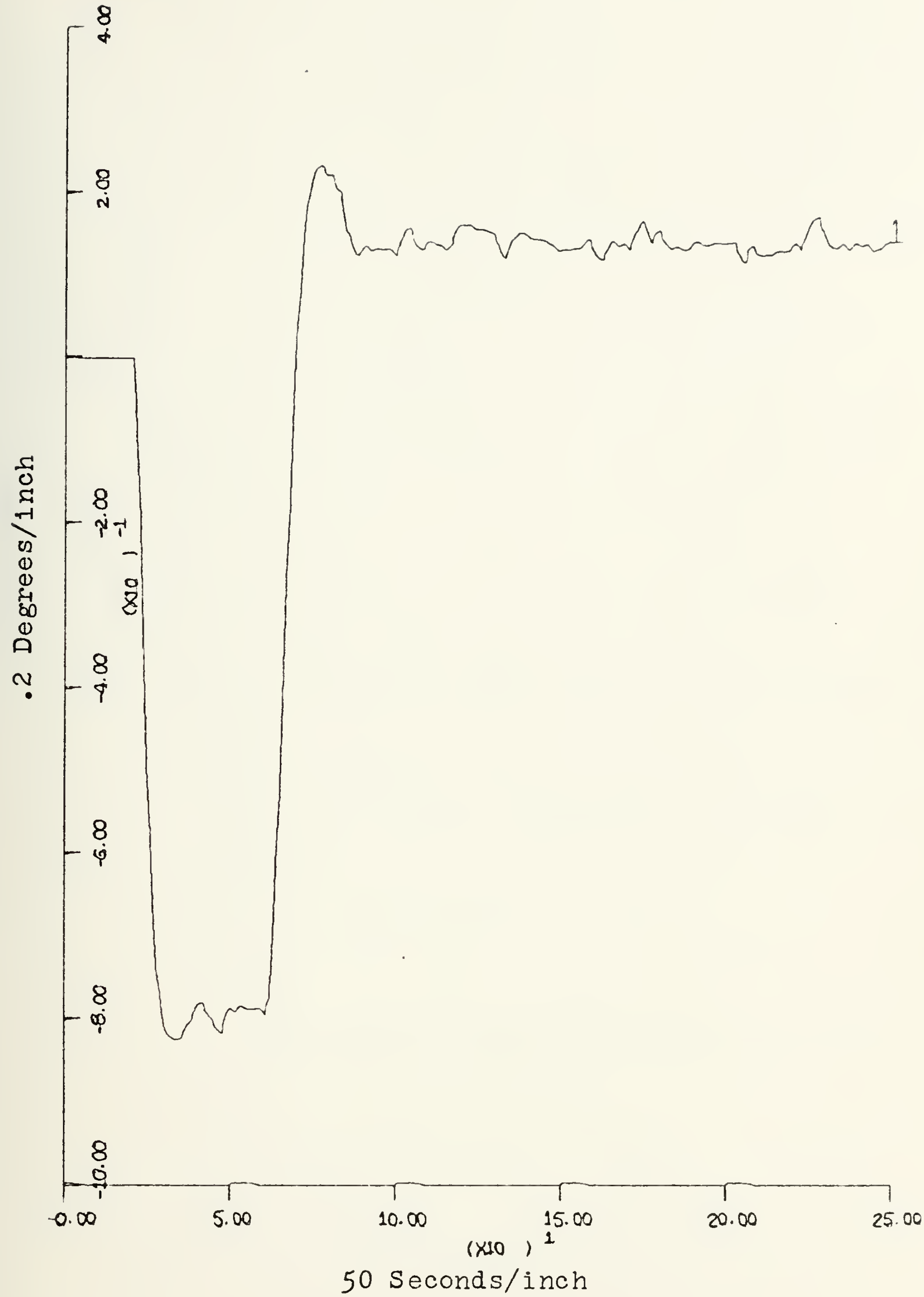


Figure III-15
Pitch vs. Time

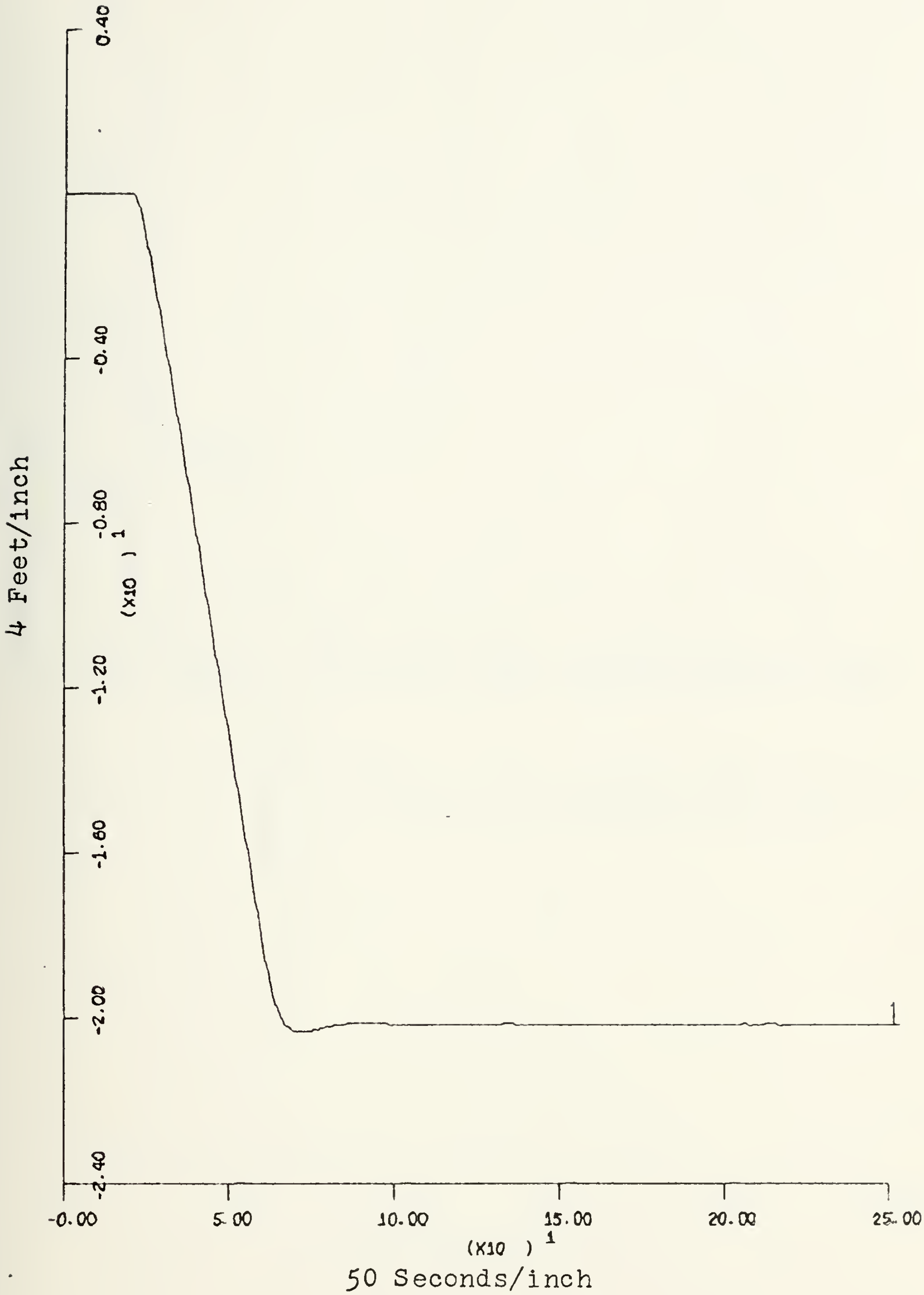


Figure III-16
Depth vs. Time

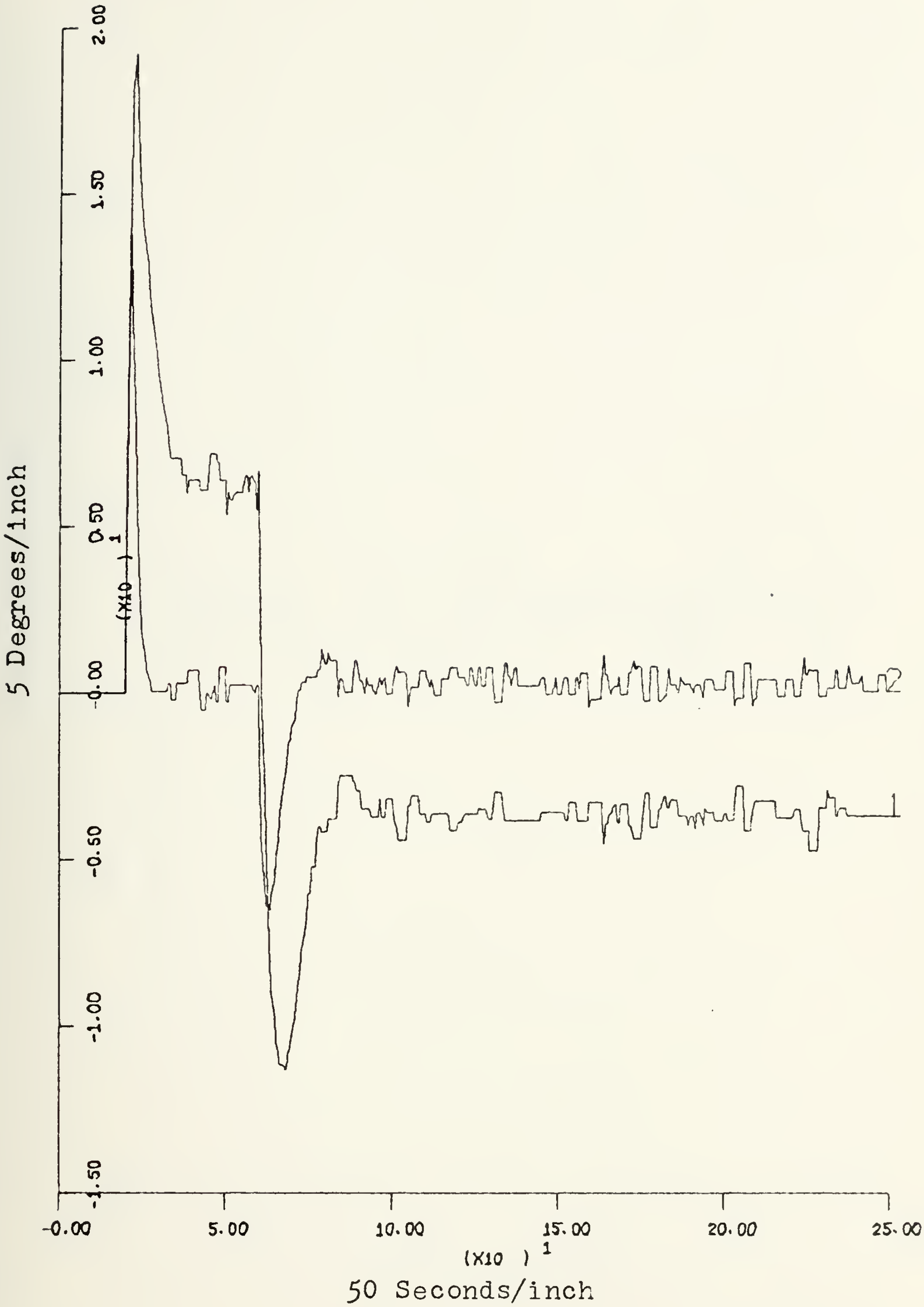
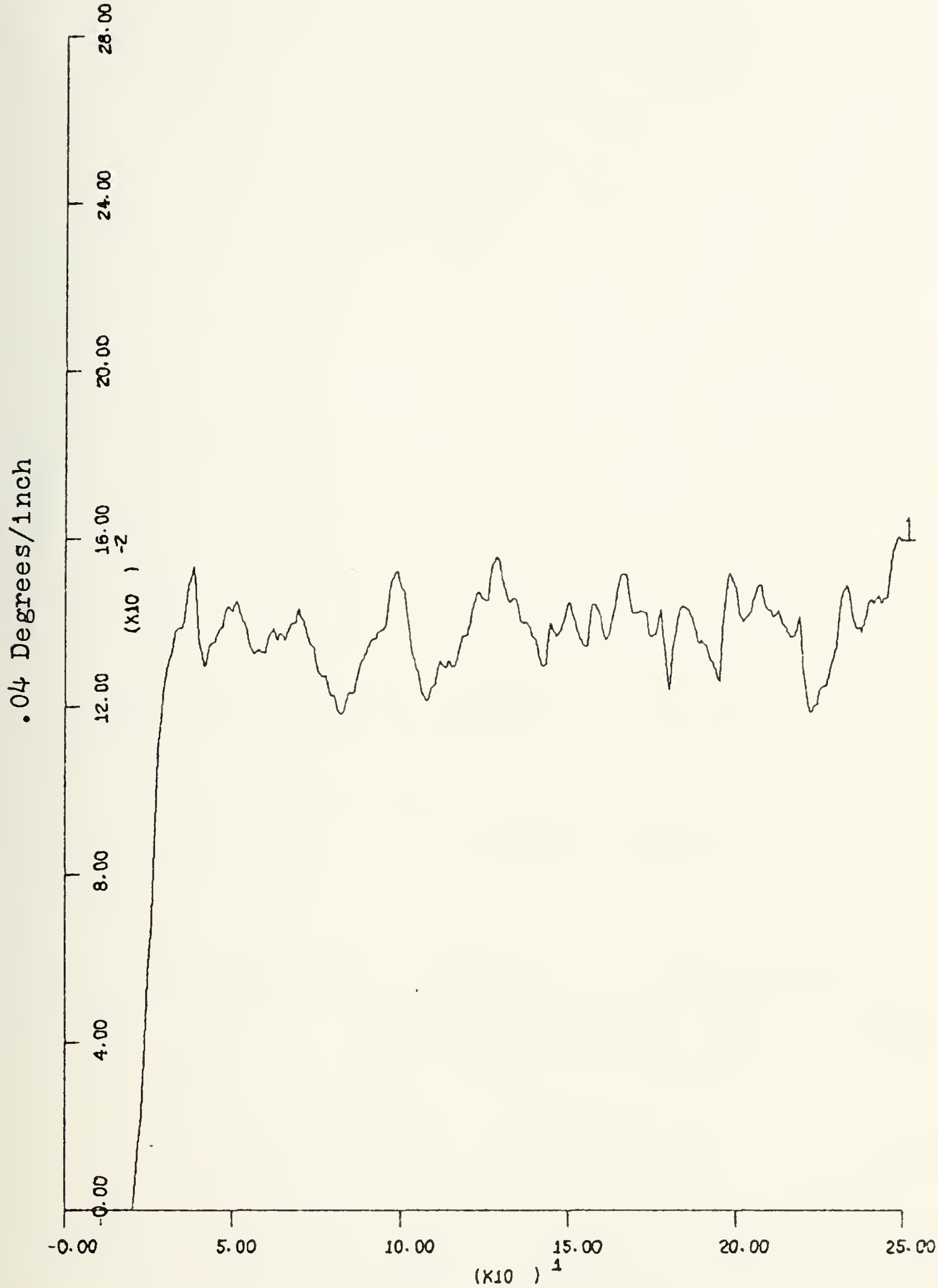


Figure III-17
Curve 1, Fairwater Planes Angle vs. Time
Curve 2, Stern Planes Angle vs. Time



50 Seconds/inch

Figure III-18
Pitch vs. Time

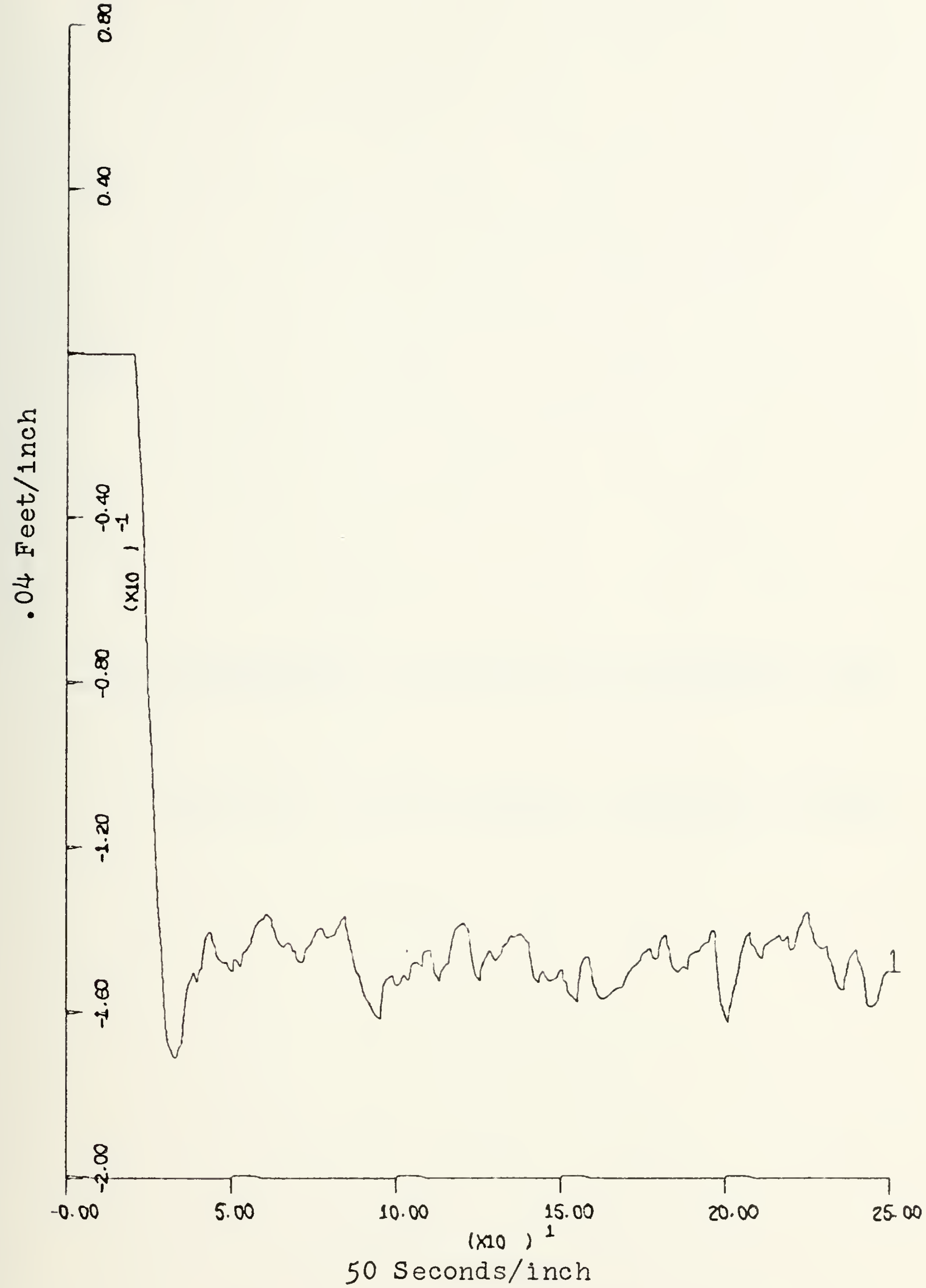


Figure III-19
Depth vs. Time

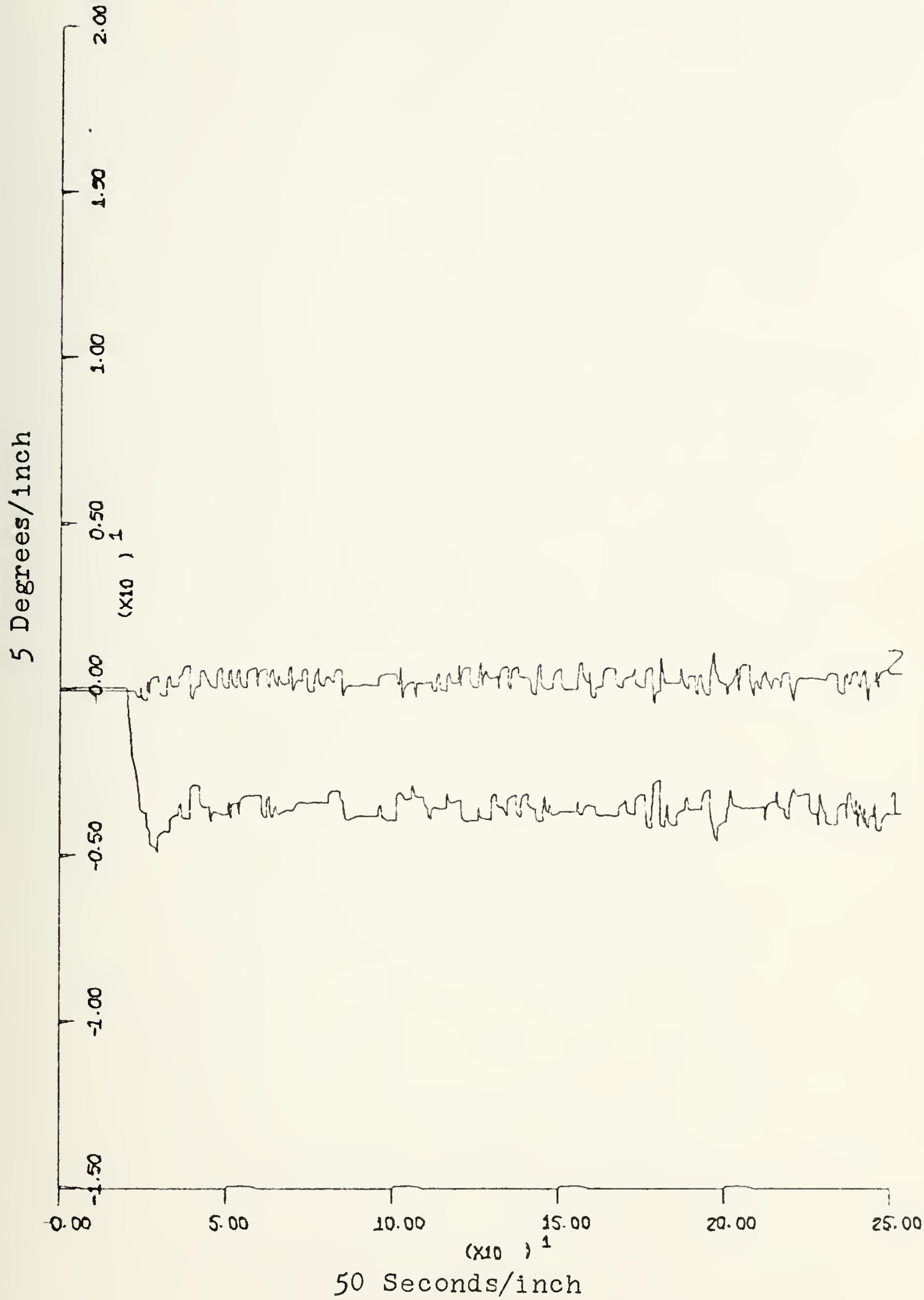


Figure III-20
Curve 1, Fairwater Planes Angle vs. Time
Curve 2, Stern Planes Angle vs. Time

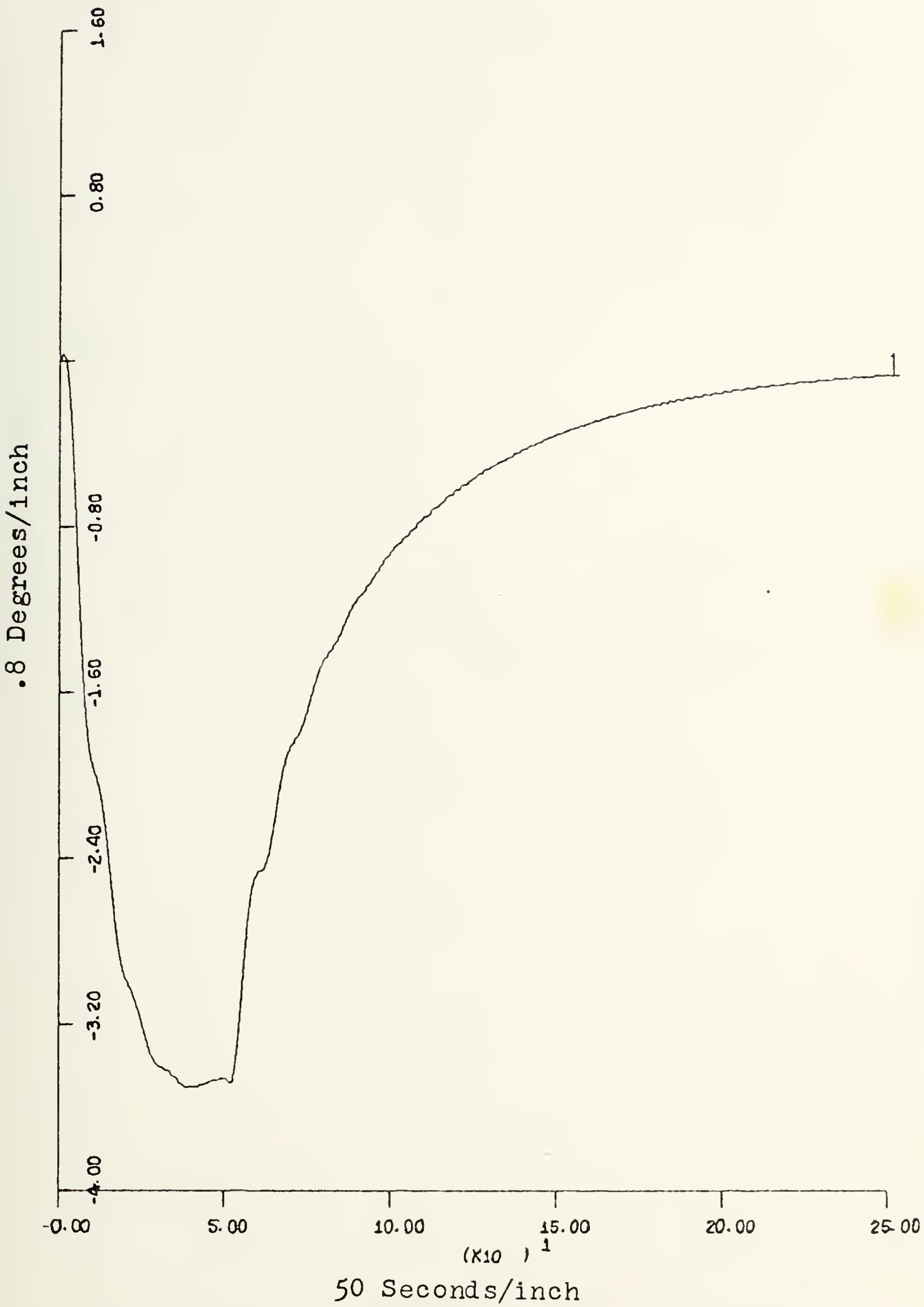
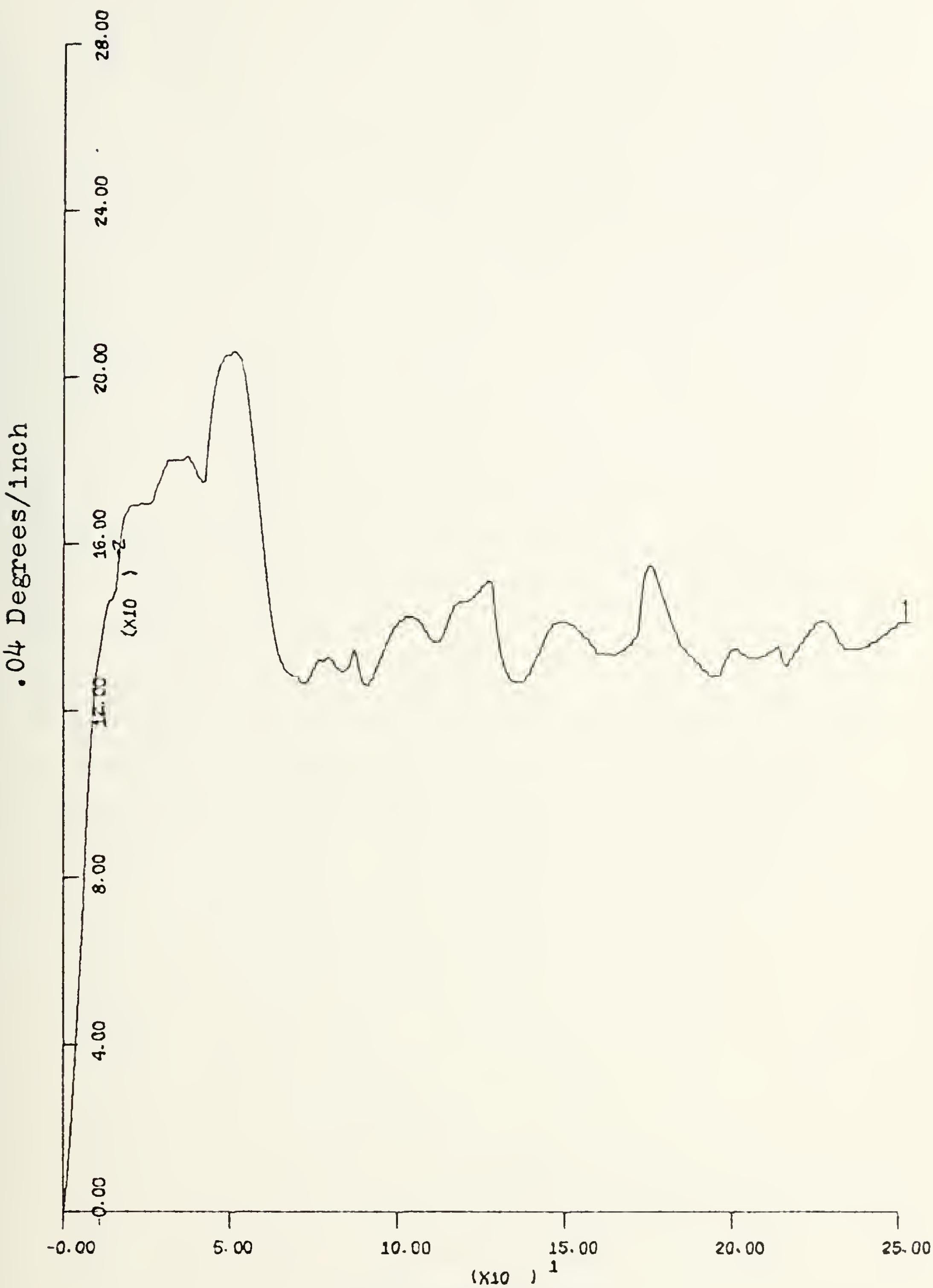


Figure III-21
Roll vs. Time



50 Seconds/inch

Figure III-22
Pitch vs. Time

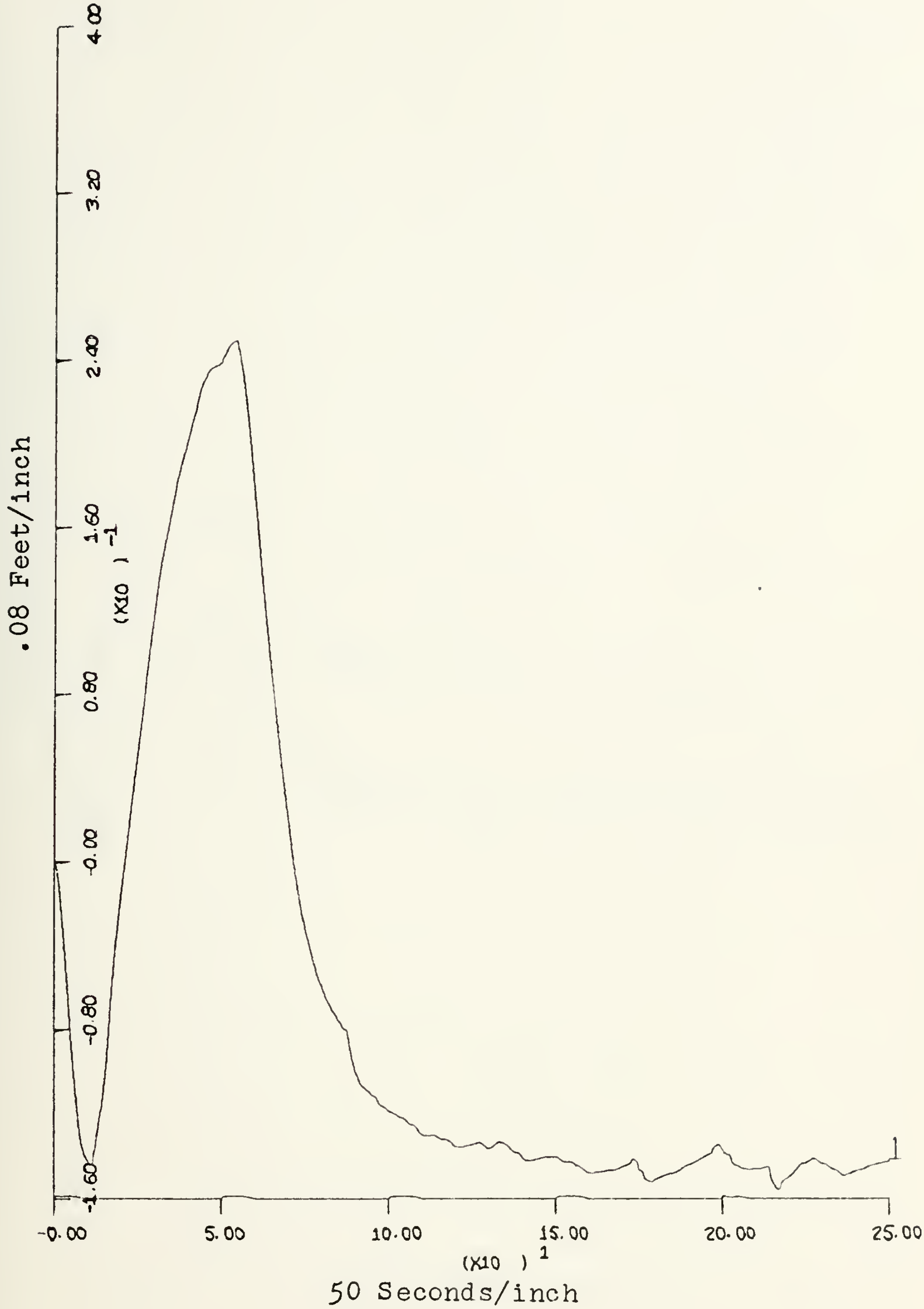


Figure III-23
Depth vs. Time

Figure III-24
Curve 1, Fairwater Planes Angle vs. Time
Curve 2, Stern Planes Angle vs. Time

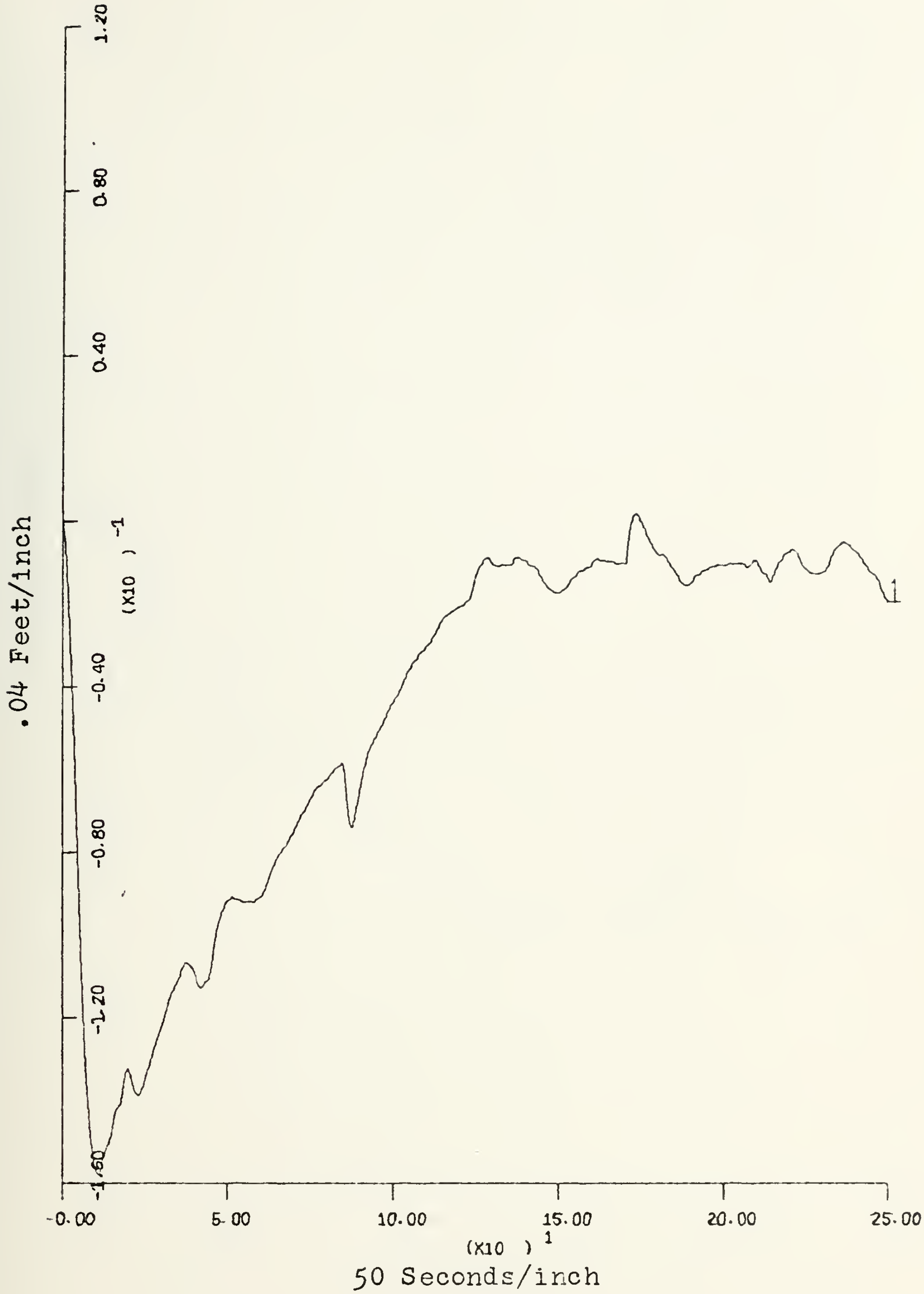


Figure III-25
Depth vs. Time

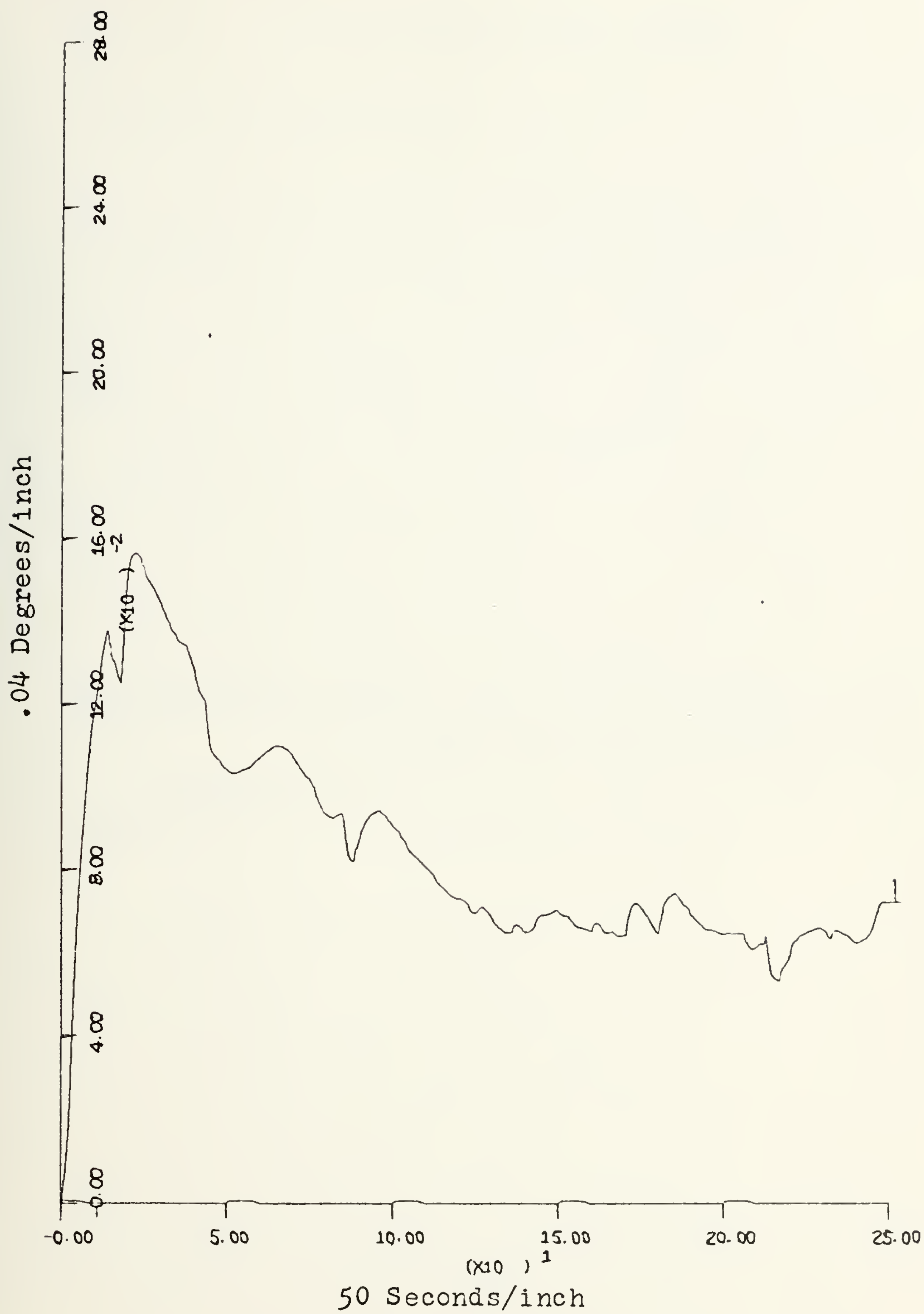


Figure III-26
Pitch vs. Time

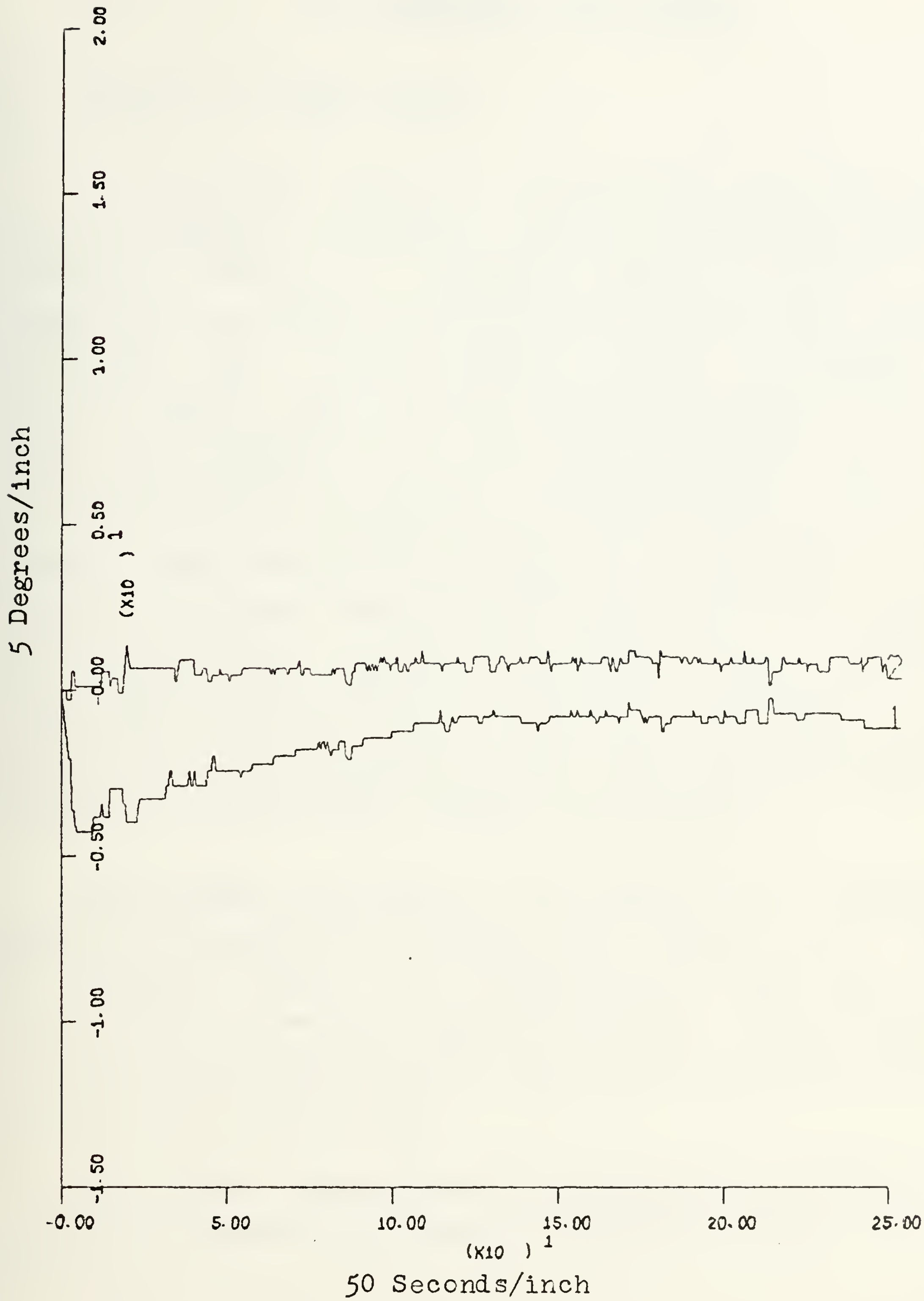


Figure III-27
Curve 1, Fairwater Planes Angle vs. Time
Curve 2, Stern Planes Angle vs. Time

IV. AUTOMATIC TRIM CONTROL

A. METHODS OF TRIM CONTROL

A continuous monitoring and control of the submarine trim is required. A submarine that is "heavy" runs with the fairwater planes on a constant positive angle. If the submarine is too heavy the fairwater planes angle reaches saturation and a large positive pitch is required to maintain depth. The speed of the submarine along with the angle of inclination of the planes determine the lifting force developed by the planes. At a high speed a submarine that is "heavy" may be well within the control limits of the planes. When the speed is decreased the decreased lifting forces would cause larger plane angles to be ordered until a possible saturation of the planes could occur. A submarine at any speed with a constant steady state fairwater or stern planes angle is experiencing a drag that is reducing the efficiency of the propulsion plant. These conditions must be corrected to efficiently operate the submarine.

Two options for automatic trim control were considered. One was to determine the steady state lifting forces caused by excessive ballast and display the magnitude of the ballast error allowing a manual correction. The next was an automatic controller in the loop pumping or moving ballast as needed to maintain trim.

The first option would be accomplished by solving the linearized equations developed in previous sections in the form

$$\text{Force} = AW + BQ + CDs + EDb$$
$$\text{Moment} = AW + BQ + CDs + EDb$$

Where in the steady state, if the plane angles are non zero the forces and moments are equivalent to the weight of required ballast to be moved. This option was not explored.

The second option was accomplished using a model following technique. The velocities of a linearized model running in a computer and those of the submarine are compared as in fig. IV-1. Both the model and the plant have the same plane angle inputs. For zero plane angles and operation on ordered depth the velocities in the vertical direction and around the Y axis for both the plant and model are zero. Perturbations caused by the planes to achieve a new ordered depth should cause both the plant and the model to accelerate in the same direction. However if a steady state plane angle is required to maintain depth or pitch in the submarine these plane angles represent accelerations in the model. A comparison of the submarine which is at nearly zero velocity and the model which is accelerating would indicate a ballast differential between a model in "trim" and a submarine out of "trim". There will be some error in the model but for small perturbations the error should be small. The model actually used was the same as used in previous sections for the derivation of the depth controller.

The need for the watch officers to maintain absolute control of the ballast shifted in or out of the ship would prevent an automatic trim controller from working in other than a passive override mode. In this mode the controller signals the operator that corrective action must be taken to maintain the proper trim. The operator, if conditions permit will actively allow the controller to perform its task. With no action on the part of the operator the system would not operate.

B. MODEL COMPARISON FOR TRIM CONTROL

In all the simulations that follow the velocities used are W and Q. The error resulting is small as long as pitch and roll are small. It was observed that with automatic control the pitch was kept relatively small. The effect of the roll will be discussed later. All runs are made with the submarine in an automatic depth control mode.

To get an idea of the magnitude of the velocities involved the first runs consisted of comparing the model and submarine velocities in the W and Q direction while manually initiating a shift of ballast. In fig. IV-2 ballast was pumped into the auxiliary tank. The bottom curve is the fairwater planes angle which moved towards a zero position. This is the same result observed in the previous section. The top curve is the stern planes angle and the middle curve is 10 times the velocity error. The error signals are multiplied to fit them on the same plot as the planes. Note that the velocity error crossed zero at approximately the same time as the fairwater planes were at zero. This was the desired response to adjust trim for a zero fairwater plane angle.

This simulation run was repeated, this time ballast was shifted from after trim tank to forward trim tank. In fig. IV-3 the fairwater planes angle and stern planes angle, curve 1 and 2, were adjusted to overcome the turning moment caused by the ballast shift. Curve 3 is 2000 times the velocity errcr in the Q direction. This curve crossed zero at about the same time as the stern plane angle crossed zero. This was the desired response for a zero stern plane angle when the suomarine was in "trim".

The controller used was an on-off controller. The pump operated at a fixed rate. The error detectors were to sense

an error, determine the direction of the error and then operate the pump. To prevent the pump from operating continuously a deadzone was inserted in the error signal channel. the deadzone was used to limit cycling of the pump during transients but its magnitude was constrained by the plane angles allowed in steady state. Adjustment of the deadzone would also determine the frequency of pump cycling and the steady state accuracy.

In fig. IV-4 the vertical velocity error and the steady state accuracy were plotted with a history of maneuvers as follows. At time zero the rudder was put at a left 10°, in 50 seconds it was moved back to zero. At 100 seconds the depth ordered was -20.0 feet. The peak observed at time 70 seconds would prevent any fine control of trim. This peak was the result of the settling of the submarine in a turn as observed in previous sections. The settling or squatting is compensated for with plane angle and these plane angles accelerate the model. This is not a serious poblem and can be solved in a number of ways. One way would be to increase the degrees of freedom in the model. This would of course be a more difficult model to construct in a computer. Alternately the system can disengaged in a turn. This would reduce the capability of the system by a small degree. Since the time spent in actually turning is small disengaging might be the best course of action. For use in this thesis it was assumed that this second course of action was taken. The deadzones were then selected at .0008 radians/second for rotational velocity error and .08 feet/second for verical velocity error.

The controller had two separate modes. One was to move ballast in and out of auxiliary tank to sea. The other was to move ballast between forward and after trim tanks. These modes were first checked out separately. First the

ballast the pump was shifted to pump from auxiliary for 79 seconds. At this point the controller stopped pumping out of the auxiliary tank and again pumped from after trim to forward trim for 20 seconds and then the pump stopped. 70 seconds after the pump stopped it started and pumped for 20 seconds out of the auxiliary tank. The end result on the steady state angle of the fairwater planes was $.6^{o}$ and $.2^{o}$ on the stern planes. Figure IV-9 shows the tank levels as ballast is moved to achieve a neutral trim. Figure IV-10 and 11 plot the planes angles. The planes move toward a zero steady state angle and a neutral trim is approached.

To denomstrate the effect of the deadzone the vertical velocity error deadzone was reduce to .04 feet/second from .08 feet/second. Figure IV-12 is the tank levels where the initial out of trim condition was the same as for fig. IV-9. A comparison of fig. IV-10 and 11 with fig. IV-13 and 14 shows the steady state planes angles were closer to zero with the smaller deadzone. Figure IV-15 and 16 show the depth controller while ballast is shifted. Pitch is controlled to $.02^{o}$ error and depth to a .1 foot error.

When the ship was ordered to change depth by 20 feet with the ship in trim the model followed closely enough so that no pump commands were given.

An additional feature in the system was tested. On some maneuvers it is desirable to operate the submarine "heavy". This is normally accomplished by simply adding for example 5000 lbs to the auxiliary tank. Whether or not the submarine is actually 5000 lbs heavy depends on the initial state of the trim. If the submarine is initially 2500 lbs light it is now only 2500 lbs heavy vice the desired 5000 lbs. By adding a weight forcing term to the model in the W equation and assigning a value to this forcing term when the

heavy or light conditon is desired the pump will shift water into or out of the auxiliary tank until the desired condition is reached. The accuracy of accomplishing this is determined by the deadzone in the vertical velocity error signal. For the submarine simulated a value of .02 feet/second worked nicely.

The program used for the simulation is program #2. Once the gains and cofactors are determined they can be read into the program as parameters and the subroutines and Riccati Equations (gain equations on page 38) can be removed. The controllers with filters as used is shown in fig. IV-17.

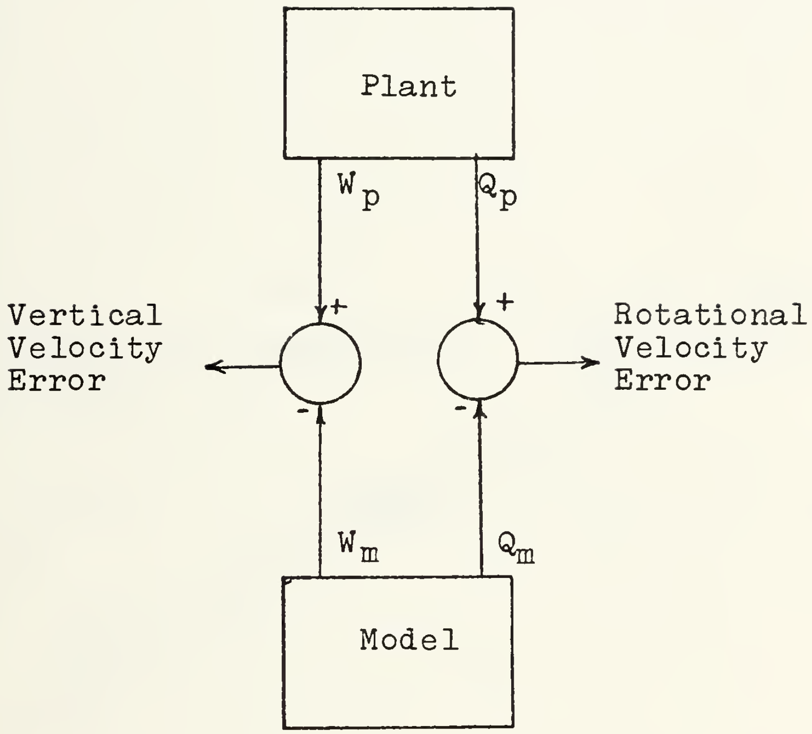


Figure IV-1

Model Comparison Block Diagram

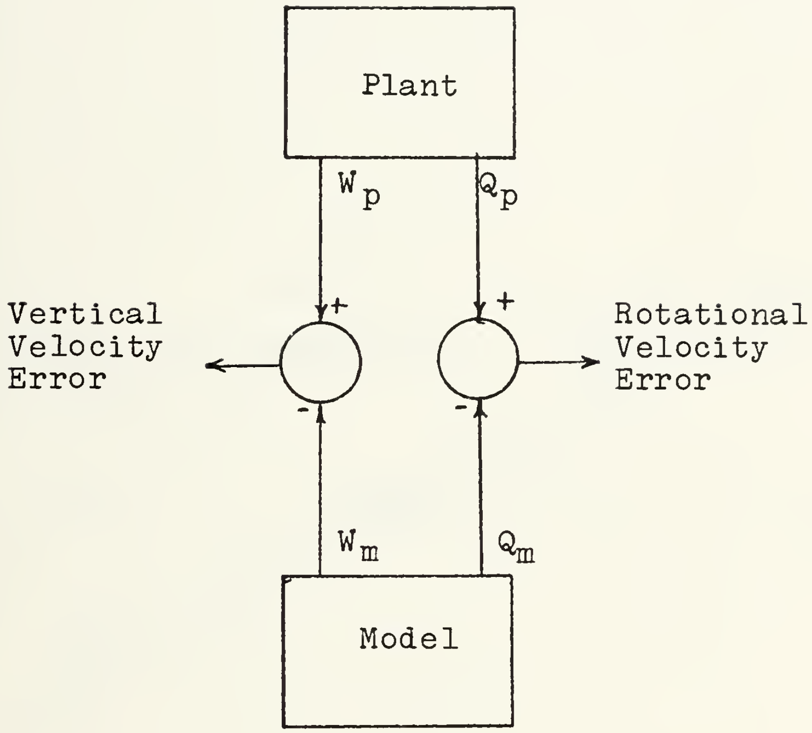


Figure IV-1

Model Comparison Block Diagram

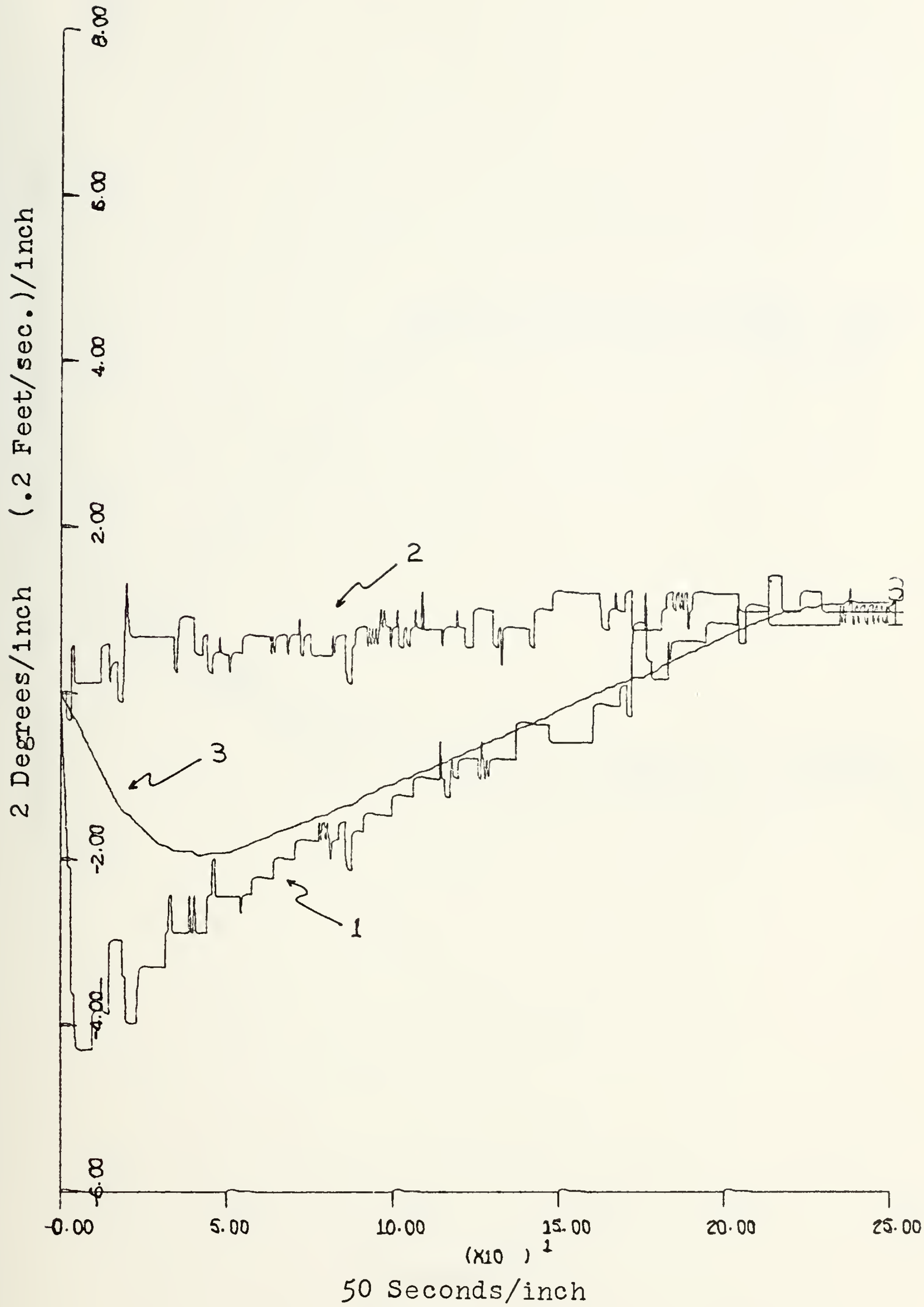


Figure IV-2
Curve 1, Fairwater Planes Angle vs. Time
Curve 2, Stern Planes Angle vs. Time
Curve 3, Vertical Velocity Error vs. Time

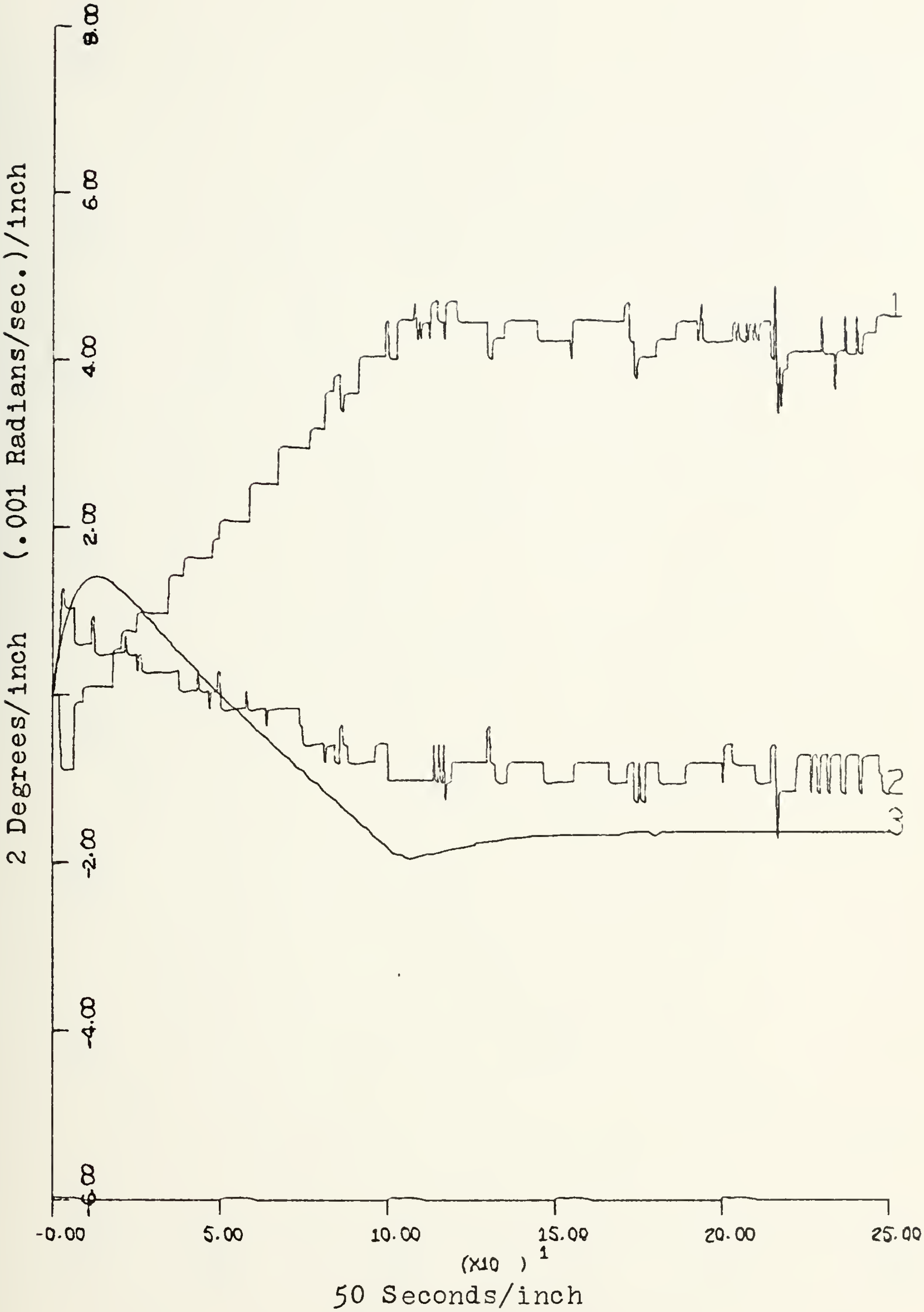


Figure IV-3
Curve 1, Fairwater Planes Angle vs. Time
Curve 2, Stern Planes Angle vs. Time
Curve 3, Rotational Velocity Error vs. Time

Figure IV-4
Vertical Velocity Error vs. Time

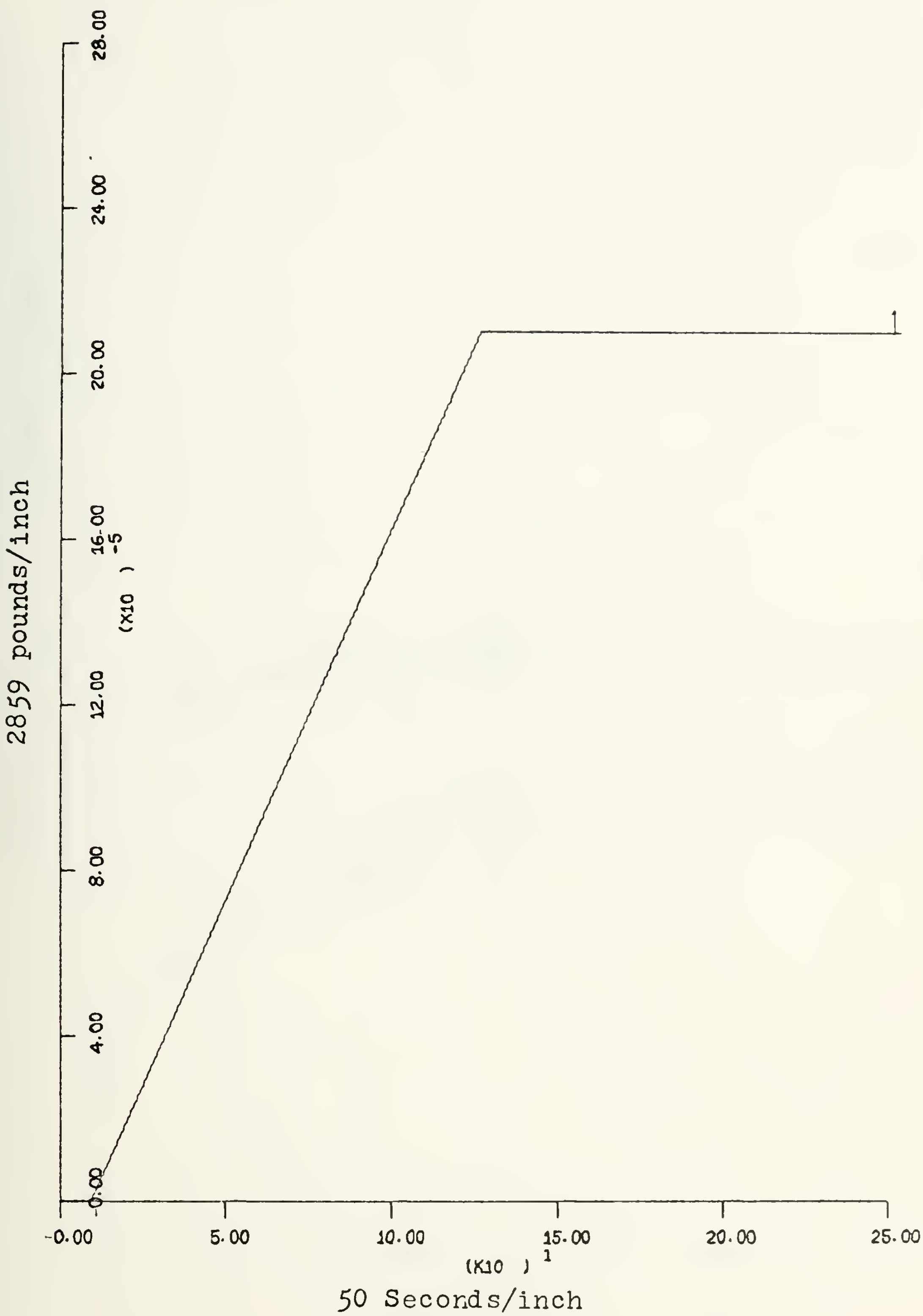
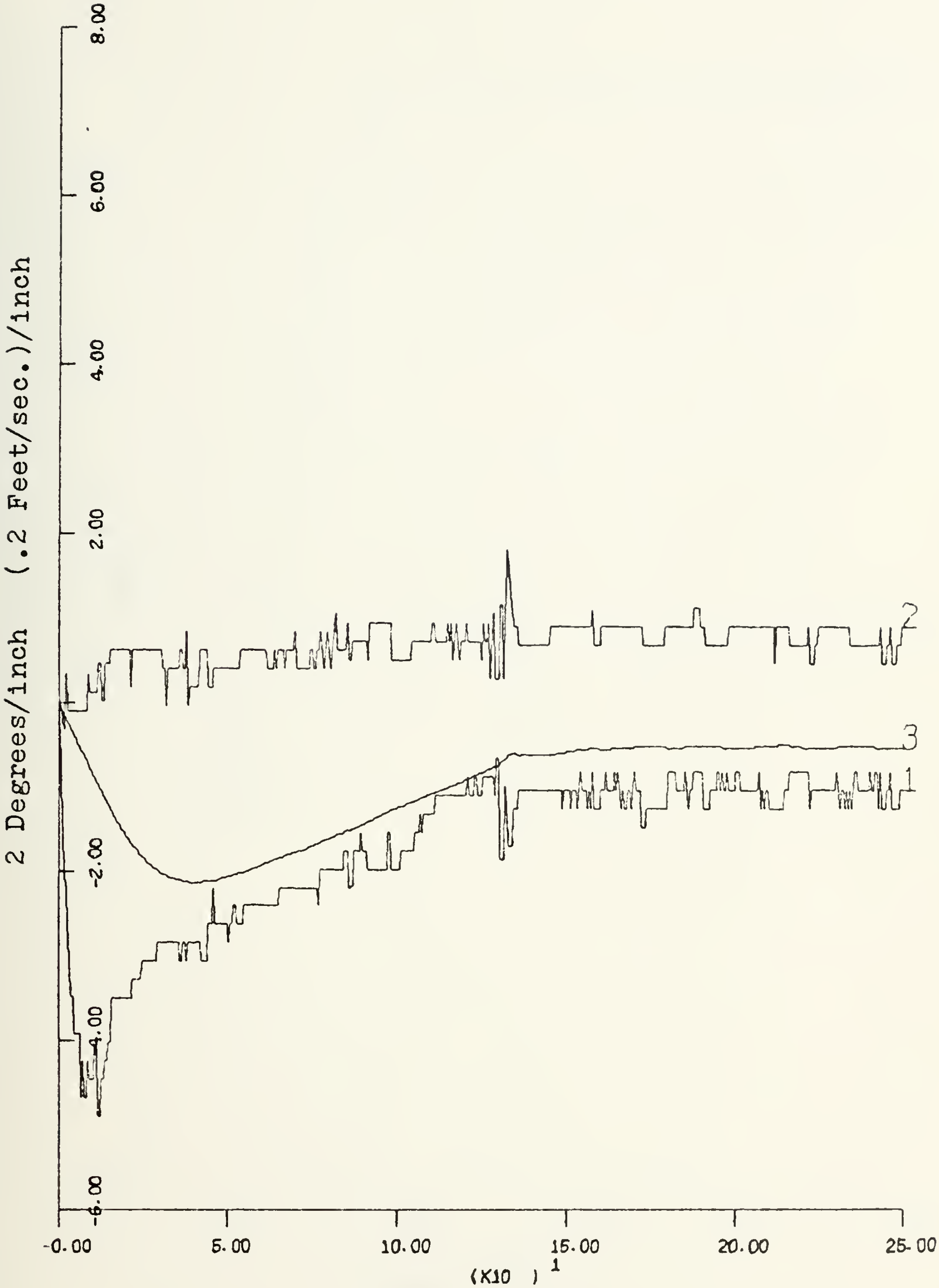


Figure IV-5
Auxiliary Tank Level vs. Time



50 Seconds/inch

Figure IV-6
Curve 1, Fairwater Planes Angle vs. Time
Curve 2, Stern Planes Angle vs. Time
Curve 3, Vertical Velocity Error vs. Time

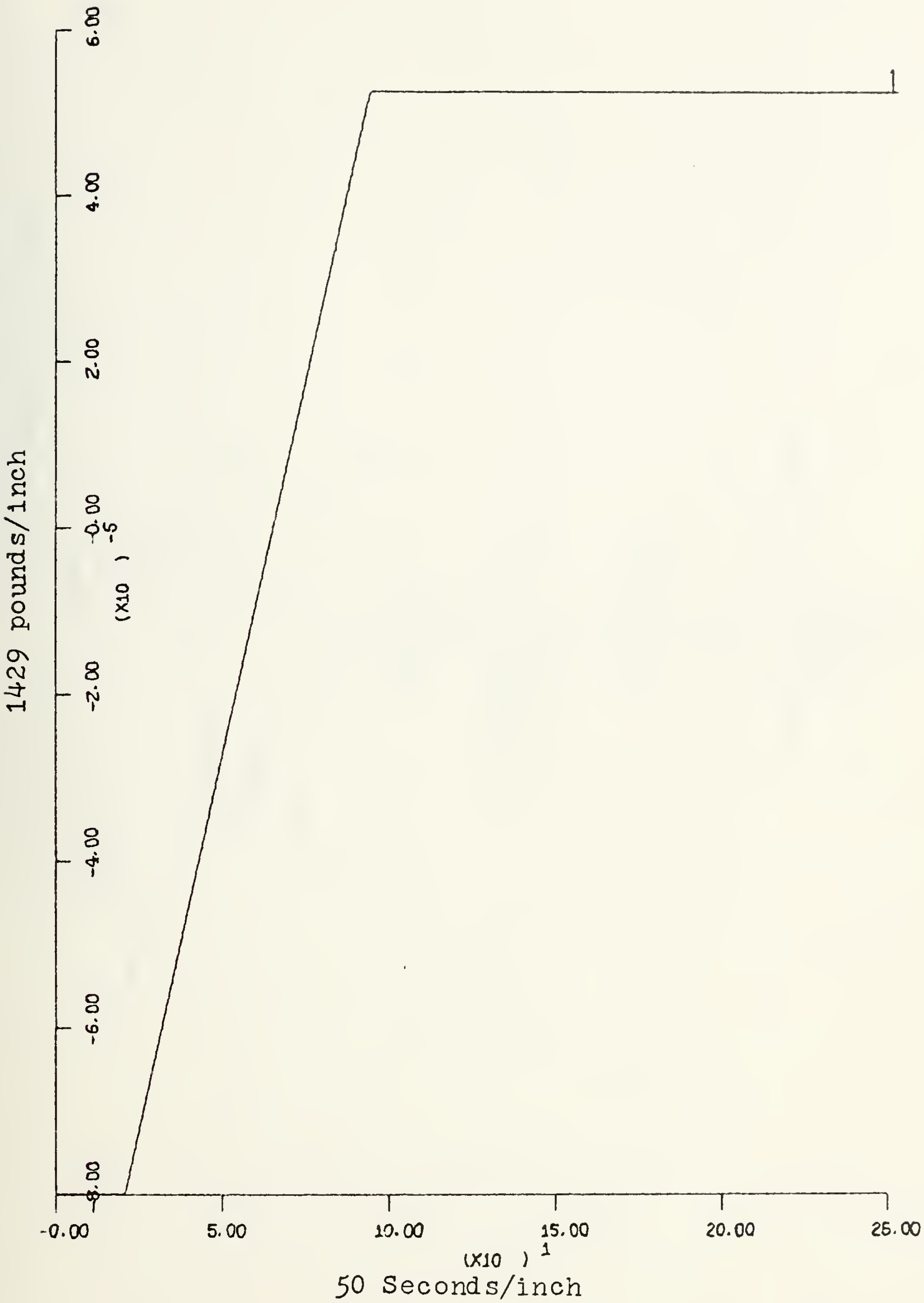


Figure IV-7
Forward Trim Tank vs. Time

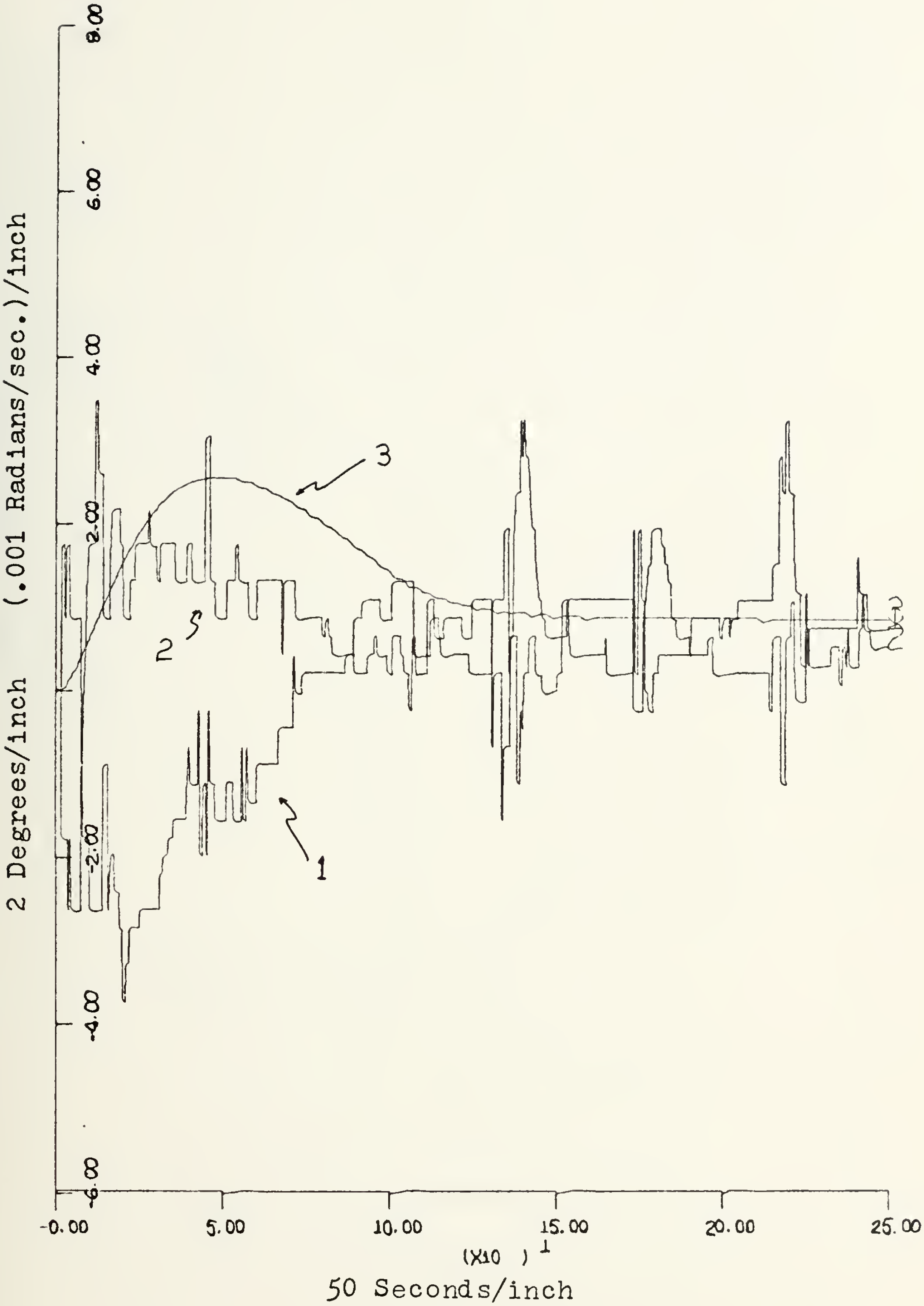


Figure IV-8
Curve 1, Fairwater Planes Angle vs. Time
Curve 2, Stern Planes Angle vs. Time
Curve 3, Rotation Velocity Error vs. Time

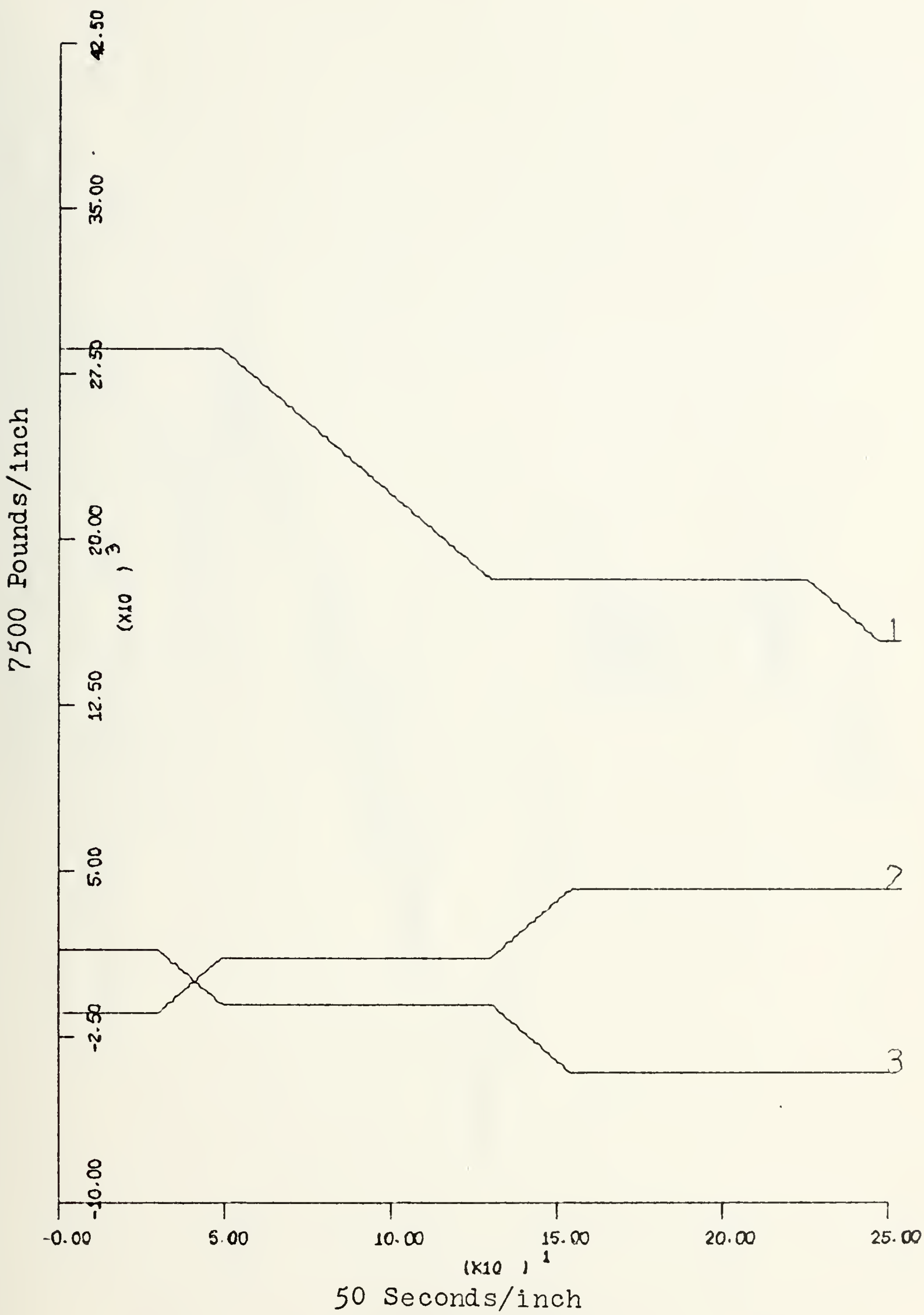


Figure IV-9
Curve 1, Auxiliary Tank Level vs. Time
Curve 2, Forward Trim Tank Level vs. Time
Curve 3, After Trim Tank Level vs. Time

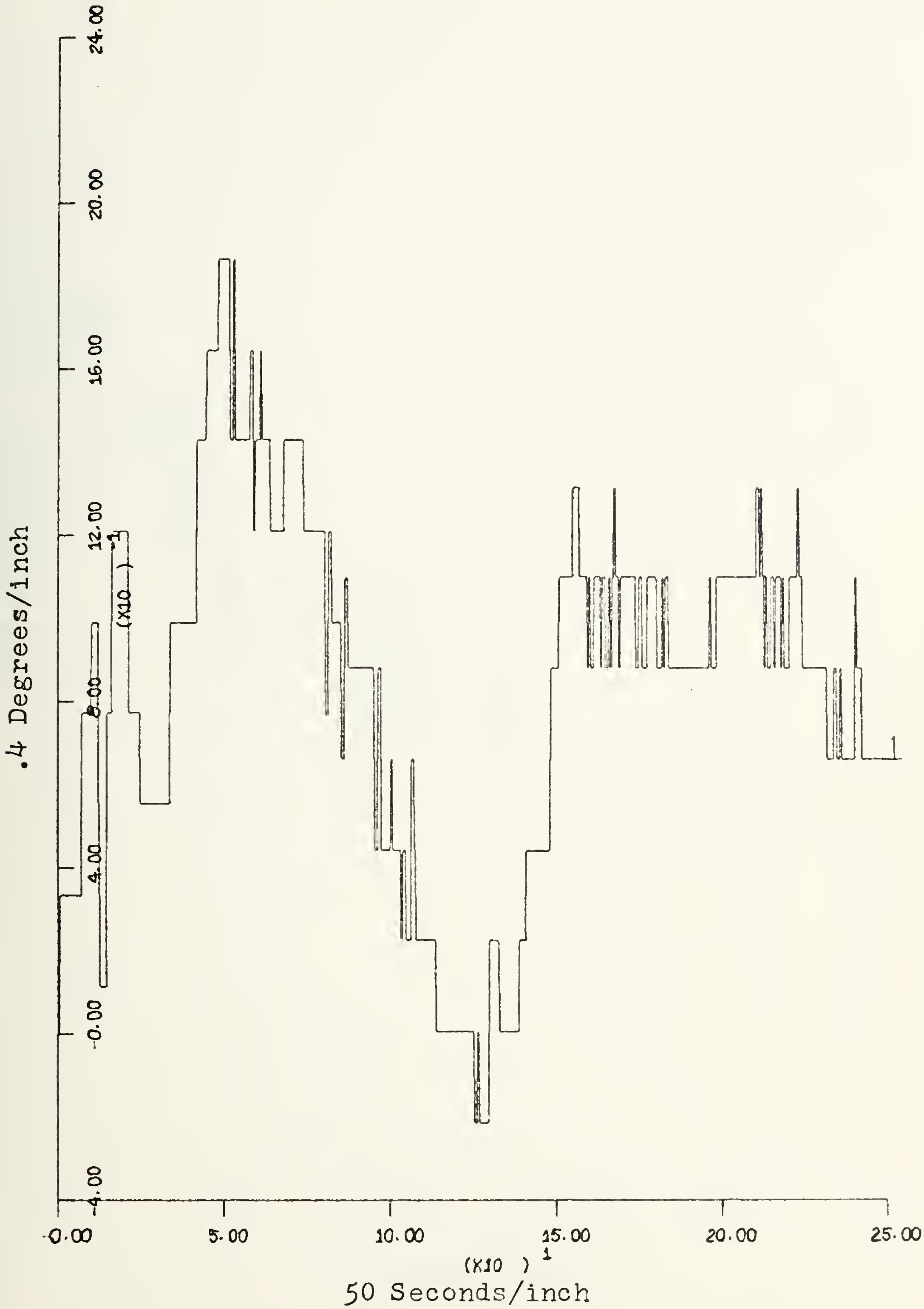


Figure IV-10
Fairwater Planes Angle vs. Time

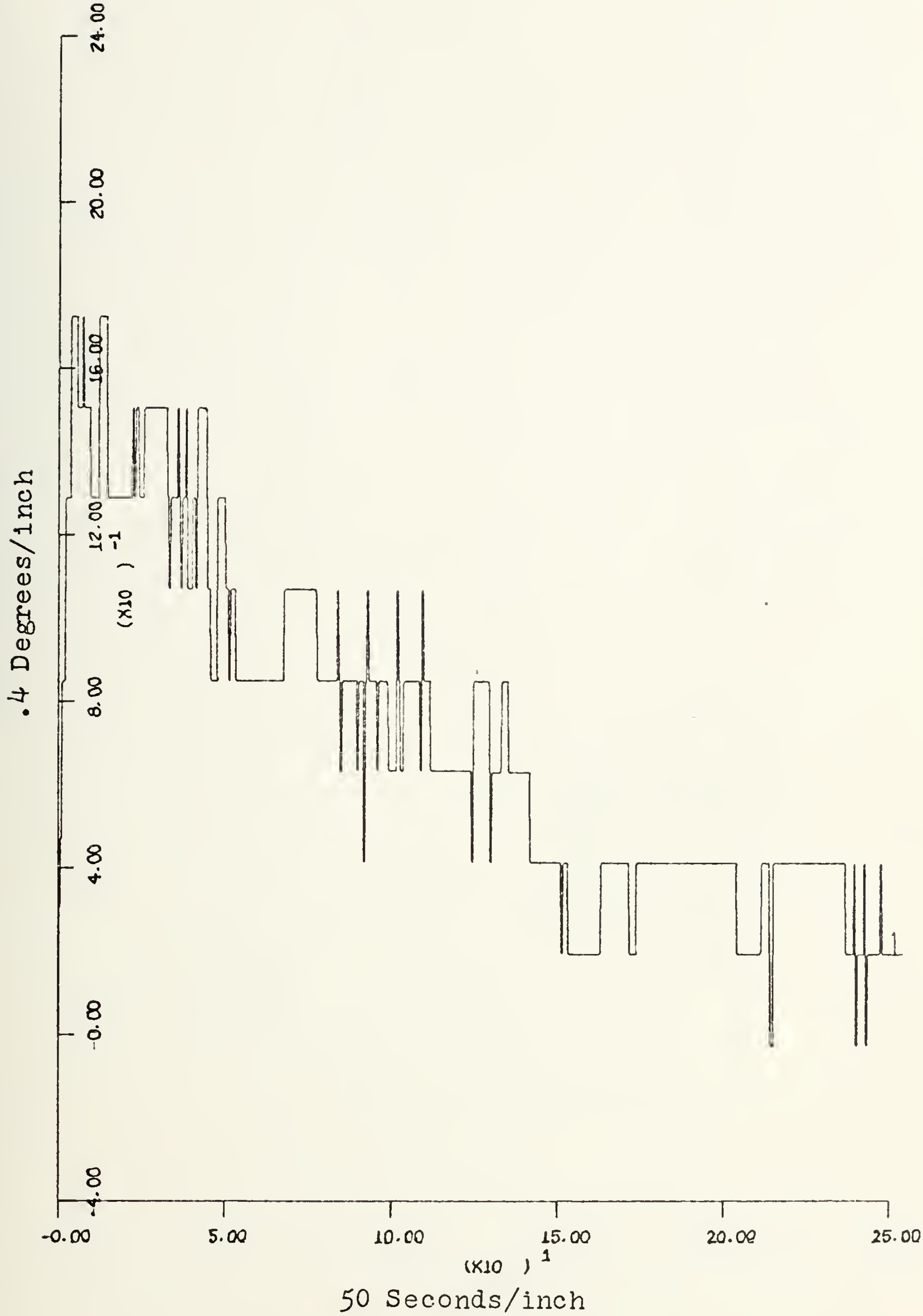
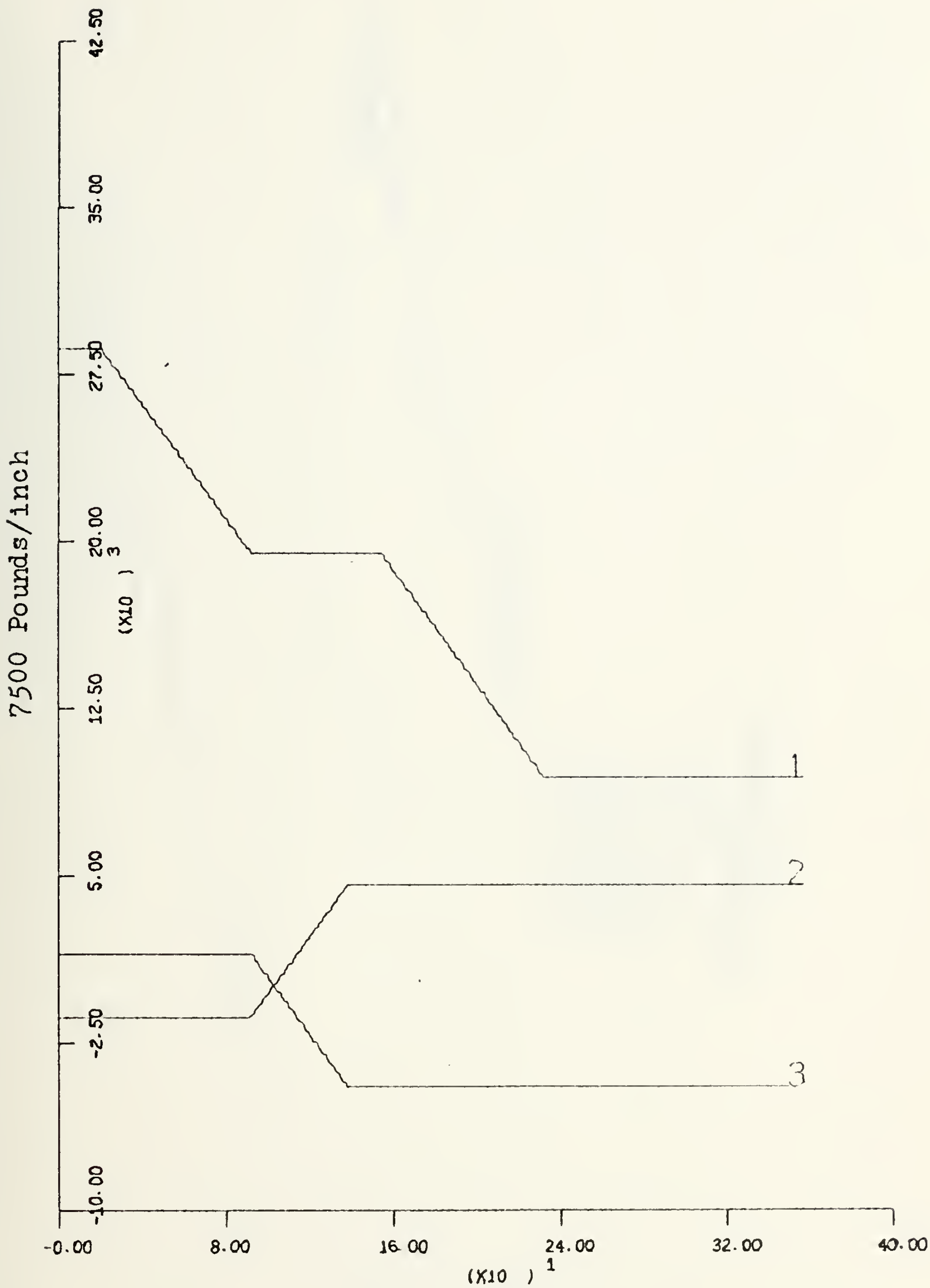


Figure IV-11
Stern Planes Angle vs. Time



80 Seconds/inch

Figure IV-12
Curve 1, Auxiliary Tank Level vs. Time
Curve 2, Forward Trim Tank Level vs. Time
Curve 3, After Trim Tank Level vs. Time

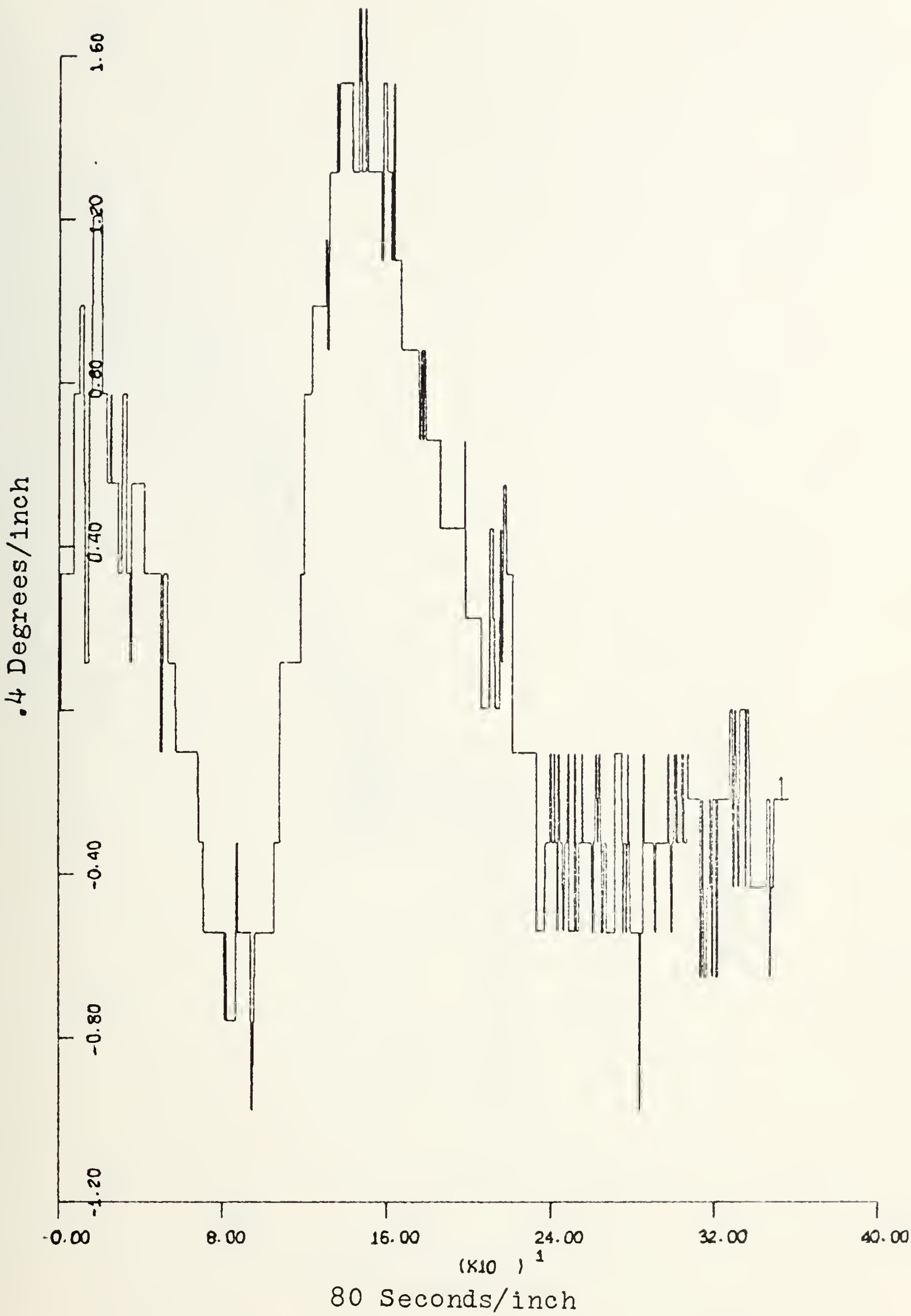


Figure IV-13
Fairwater Planes Angle vs. Time

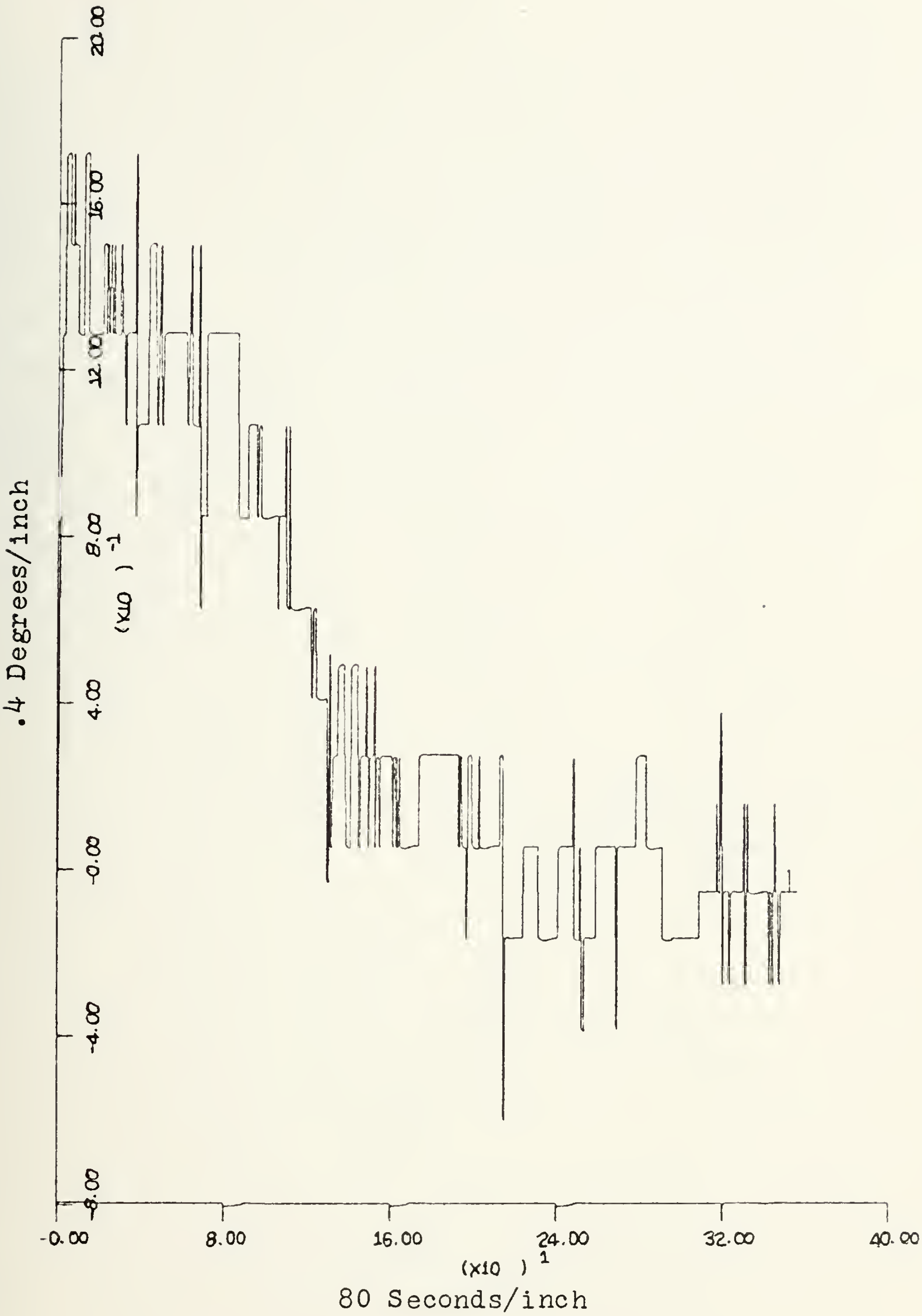


Figure IV-14
Stern Planes Angle vs. Time

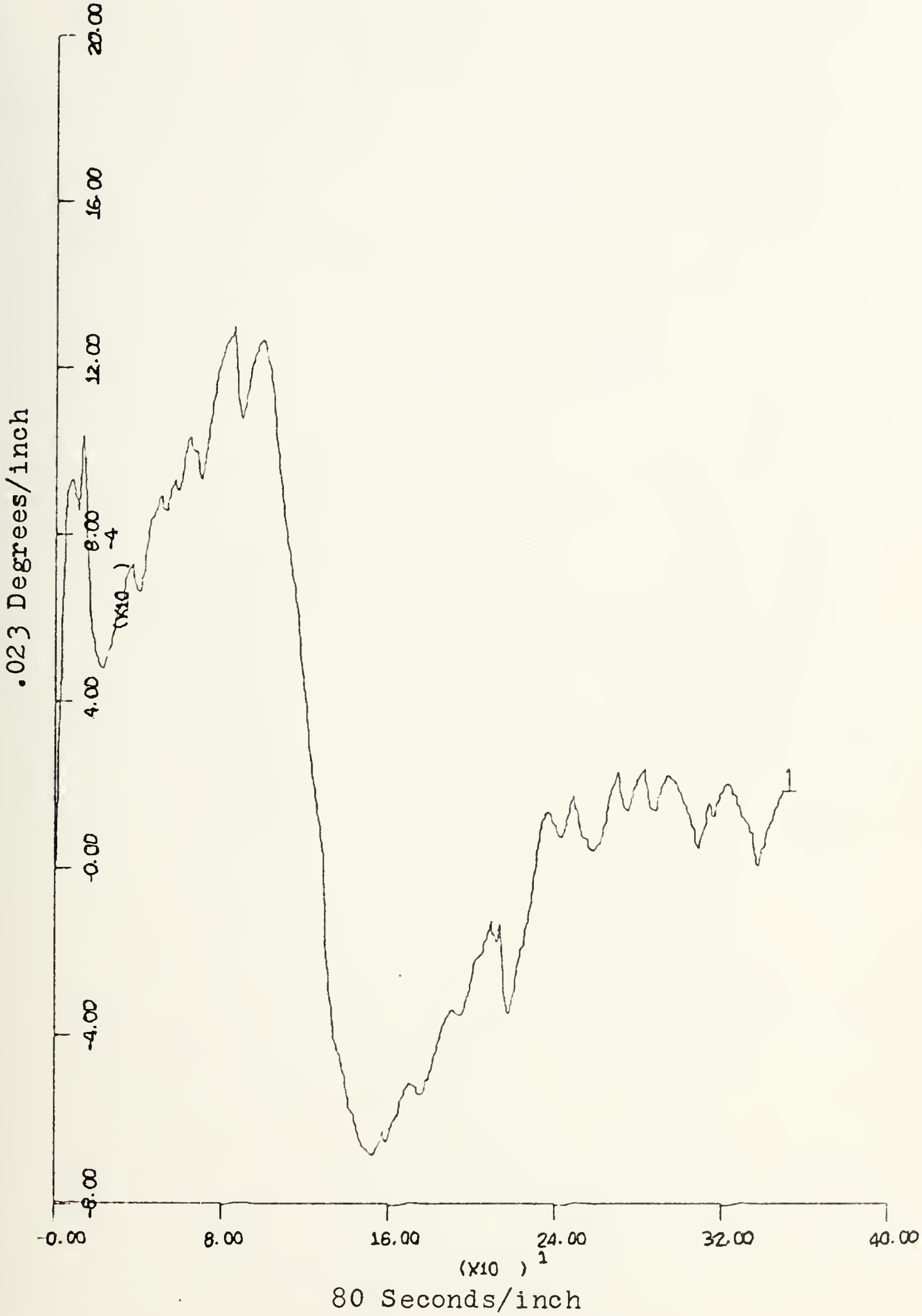


Figure IV-15
Pitch vs. Time

Figure IV-16
Depth vs. Time

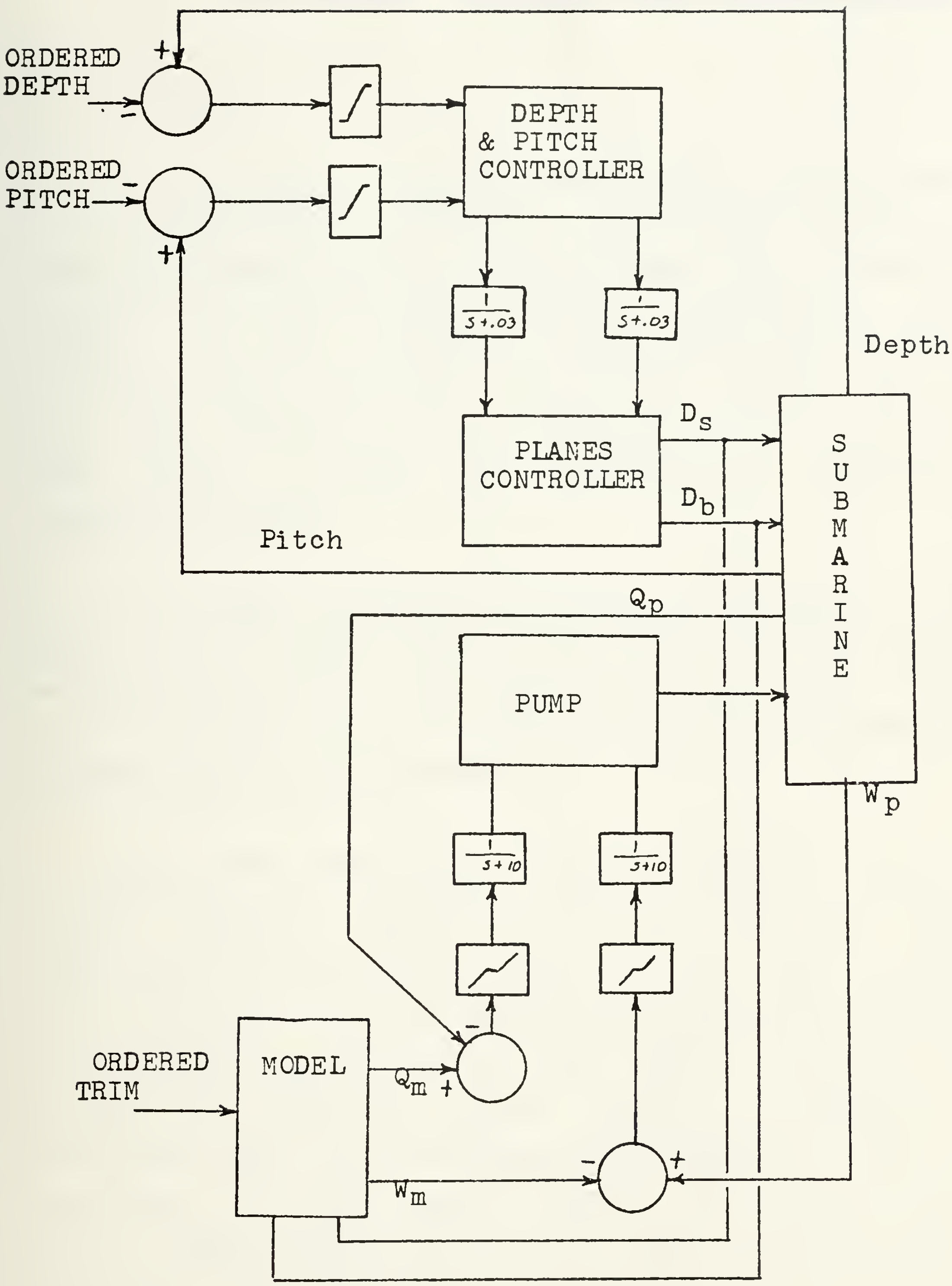


Figure IV-17

Controller Block Diagram

V. DISCUSSION, CONCLUSIONS AND RECOMMENDATIONS

A. DISCUSSION

In the design of optimal controllers a major difficulty is in determining a weighting matrix that produces the desired response. With the use of computers many simulations can be made and the weights adjusted to obtain the response needed. In this thesis no attempt was made to weight the velocities with the errors. This is an option that could give more control over the output response, but this would also greatly increase the difficulty in selecting the weighting matrices.

The optimal controller designed gave excellent results and produced steady state errors that were smaller than can be normally measured. The magnitude of the error achieved is a function of the weighting matrix selected and the forcing inputs to the system.

The depth controller was not designed as a depth changer, but as a regulator. However it did function for depth changing inputs. The depth changing rate was determined by a saturation limit in the depth and pitch error inputs . The limiter is required because the planes have a finite capability and the use of optimal control assumes an infinite control input is available. The saturation limit can be adjusted for different plane angle responses to large depth changes. This would change the rate of depth change and the pitch angle assumed during the transient. The magnitude of the saturation limit chosen was based on limiting the planes angles when a new ordered depth was given and further study of any range of limits was not made.

Trim control was accomplished using a model comparison technique. The results of this technique were in some respects better than expected. The trim controller was accurate in the final results and its corrective action was achieved earlier than a manual operator could accomplish because the error was detected sooner. The model was run in two degrees of freedom. If the model is left at two degrees of freedom something must be done to compensate for the effects observed in a roll. The system would not reliably detect a change in trim in a turn and its ability to detect a change in trim during a large vertical transient is questionable. However, the controller would not mistake a vertical transient as a demand or need for ballast change. Because of this last result it was not necessary to compensate or change the system in a dynamic vertical situation. The trim controller was not based on the depth controller used so it would conceivably work with any depth controller as long as the planes were used to compensate for pitch and depth errors. Filters would be required to give continuous average plane angles.

Using the trim controller to adjust the submarine to a "heavy" or "light" condition was not part of its intended function but the controller was able to accomplish this with reasonable accuracy. The accuracy was largely determined by the dead zone in the signal channel. The deadzone would have to be large enough not to mistake noise or ordered depth transients for a trim error. This contraint imposes a limit on the accuracy that could be achieved. The priority assigned in adjusting trim was arbitrarily assigned to first correct an overall weight problem then adjust for a fore and aft trim.

B. CONCLUSIONS

Optimal control techniques can be used to design a depth controller that achieves a steady state condition as accurately as desired. Adjustment of the weighting matrices would emphasize those errors that are most important and also adjust the amount of response by the planes to accomplish the control. The insertion of nonlinearities (in this case a saturation limit) would allow the controller to give a reasonable response to any input and provide for a depth changing capability as well as the depth maintaining capability designed for.

Steady state gains were used instead of time varying gains, and this did not degrade the functioning of the controller but did provide a significant simplification in the controller. A controller designed with the methods described would provide a vast improvement over manual operation. The gains provided are only a function of speed which makes realization of the controller relatively simple. The controller output itself is also a function of pitch rate and depth rate. This provides an anticipatory feature to the controller.

In the past trim control has been purposely left to manual control. This was for safety reasons and because the detection of an out of trim condition was a matter of judgement based on the observation of plane positions and therefore only good in the steady state. The passive override mode discussed in previous sections removes the safety constraints in the use of an automatic trim controller. The simulations run demonstrate that detection of trim error can be accomplished accurately in a dynamic situation. The method still uses the planes positions for its determination but the lifting forces of both sets of planes and the turning moments of both sets of planes are

used in the determination of trim. Traditionally only the lifting forces of the fairwater planes and the turning moment of the stern planes were considered for the trim error estimate. The controller made accurate estimates as seen in fig. IV-9 and 12 where it was noted that the controller never had to go back and correct an adjustment once it was made.

The additional feature in the trim controller of being able to deliberately order an out of trim condition provides for more positive control in the trim of the submarine. This control is a function of the actual submarine conditions and not of the assumed conditions. If the conditions change during the submarine operation such as an ascent the trim controller would automatically maintain the out of trim condition ordered. A difficult feat for a manual operator.

C. RECOMMENDATIONS FOR FURTHER WORK

There are six specific areas that need further study and work. They are enumerated in the following paragraphs.

1. The selection of the saturation limits in the error channel was arbitrary and the only criteria was to prevent the planes angles from becoming excessive in a depth changing situation. As in the actual controller there must be some optimum limit that would meet a specific criteria in changing depth. this limit could be found using a function minimization program where numerous simulation runs are made and compared to determine a cost based on meeting the criteria specified. A minimum cost would define a match to the criteria specified.

2. To achieve a zero steady state error and at the same

time make the controller an optimal depth changing controller a method of decoupling is needed. The methods of steady state state decoupling through the use of cascade compensators described in ref. 7 might provide such a controller with a function minimization procedure to meet a specified criteria. This approach would allow for zero steady state error in pitch and depth under most conditions.

3. the controllers in this thesis were designed for continuous time. To provide for use in a time shared computer environment a discrete version of the controller is required. For the trim controller this would mean a discrete model used for comparison with the submarine and a sampling of the submarine parameters.

4. The control inputs considered for the depth controller were the fairwater and stern planes. The control inputs could be expanded to include a vertical forcing function and a rotational forcing function around the Y axis. These inputs would of course be the shifting of ballast in the trim tanks. In this case the planes angles would be included as states. The purpose of this scheme is to control the planes to a zero angle by the shifting of ballast.

5. The trim controller does not work in a turn. A suggestion to correct this would be to increase the degrees of freedom of the model. The model would have to be developed and tested.

6. Finally, the options allowed for the pump were very limited and for example if the submarine is "heavy" forward and at the same time "heavy" overall the designed controller would pump ballast from auxiliary tank then pump ballast from forward to aft. A more efficient maneuver would be to pump ballast form forward trim to sea. Even for only three

variable ballast tanks the pumping operation would become significantly more complicated.

EQUATIONS OF MOTION

The following set of equations are referred to a body fixed system of axes which are coincident with the principal axes of inertia of the body. The origin of this axis-system is located at the assumed center of mass of the body

Equation of Motion Along the Body Axis System x-Axis

$$m(\dot{u} - vr + wq) = \frac{\rho}{2} \ell^4 \left[X_{qq}' q^2 + X_{rr}' r^2 + X_{rp}' rp \right]$$

$$+ \frac{\rho}{2} \ell^3 \left[X_{\dot{u}}' \dot{u} + X_{vr}' vr + X_{wq}' wq \right]$$

$$+ \frac{\rho}{2} \ell^2 \left[X_{uu}' u^2 + X_{vv}' v^2 + X_{ww}' w^2 \right]$$

$$+ \frac{\rho}{2} \ell^2 u^2 \left[X_{\delta r \delta r}' \delta_r^{\ 2} + X_{\delta s \delta s}' \delta_s^{\ 2} + X_{\delta b \delta b}' \delta_b^{\ 2} \right]$$

$$+ \frac{\rho}{2} \ell^2 X_{vvn'}' (n' - 1) v^2$$

$$+ \frac{\rho}{2} \ell^2 X_{wwn'}' (n' - 1) w^2$$

$$+ \frac{\rho}{2} \ell^2 u^2 X_{\delta s \delta s n'}' (n' - 1) \delta_s^{\ 2}$$

$$+ \frac{\rho}{2} \ell^2 u^2 X_{\delta r \delta r n'}' (n' - 1) \delta_r^{\ 2}$$

$$- \Sigma W_i \sin \theta$$

$$+ (F_x)_P$$

Equation of Motion Along the Body Axis System y-Axis

$$m(\dot{v} - wp + ur) = \frac{\rho}{2}\ell^4 \left[Y_{\dot{r}}{}'\dot{r} + Y_{\dot{p}}{}'\dot{p}\right]$$

$$+ \frac{\rho}{2}\ell^4 \left[Y_{pq}{}'pq + Y_{p|p|}{}'p|p|\right]$$

$$+ \frac{\rho}{2}\ell^3 \left[Y_{\dot{v}}{}'\dot{v} + Y_{wp}{}'wp + Y_{v|r|}{}'\frac{v}{|v|}\left|(v^2 + w^2)^{\frac{1}{2}}\right| |r|\right]$$

$$+ \frac{\rho}{2}\ell^3 \left[Y_{r}{}'ur + Y_{|r|\delta r}{}'u|r|\delta r + Y_{p}{}'up\right]$$

$$+ \frac{\rho}{2}\ell^3 \; Y_{rn'}{}'(n' - 1)\, ur$$

$$+ \frac{\rho}{2}\ell^2 \left[Y_{*}{}'u^2 + Y_{v}{}'uv + Y_{v|v|}{}'v\left|(v^2 + w^2)^{\frac{1}{2}}\right|\right]$$

$$+ \frac{\rho}{2}\ell^2 u^2\, Y_{\delta r}{}'\delta_r$$

$$+ \frac{\rho}{2}\ell^2 u^2\, Y_{\delta rn'}{}'(n' - 1)\,\delta_r$$

$$+ \frac{\rho}{2}\ell^2\, Y_{vn'}{}'(n' - 1)\, uv$$

$$+ \frac{\rho}{2}\ell^2\, Y_{v|v|n'}{}'(n' - 1)\, v\left|(v^2 + w^2)^{\frac{1}{2}}\right|$$

$$+ \frac{\rho}{2}\ell^2 \; Y_{wv}{}'wv\#$$

$$+ \frac{\rho}{2}\ell^2 (F_y)_{vs} \quad \frac{v^2 + w^2}{U} \quad (-w)\ \mathrm{Sin}\,\omega t$$

$$+ \Sigma W_i \sin\phi\cos\theta$$

\# Multiplied by $\frac{u}{U}$ for large angles of attack near $-90°$

Equation of Motion Along the Body Axis System z-Axis

$$m(\dot{w} - uq + vp) = \frac{\rho}{2} \ell^4 Z_{\dot{q}}' \dot{q}$$

$$+ \frac{\rho}{2} \ell^4 \left[Z_{rr}' r^2 \# + Z_{rp}' rp\# \right]$$

Note 1

$$+ \frac{\rho}{2} \ell^3 \left[Z_{\dot{w}}' \dot{w} + Z_{vr}' vr\# + Z_{vp}' vp + \Delta Z_{vp}' vp\# \right]$$

$$+ \frac{\rho}{2} \ell^3 \left[Z_q' uq + Z_{|q|\delta s}' u|q|\delta s + Z_{w|q|}' \frac{w}{|w|} \left| (v^2 + w^2)^{\frac{1}{2}} \right| |q| \right]$$

$$+ \frac{\rho}{2} \ell^3 Z_{qn'}' (n' - 1) uq$$

$$+ \frac{\rho}{2} \ell^2 \left[Z_*' u^2 + Z_w' uw + Z_{w|w|}' w \left| (v^2 + w^2)^{\frac{1}{2}} \right| \right]$$

$$+ \frac{\rho}{2} \ell^2 \left[Z_{|w|}' u|w| + Z_{ww}' \left| w (v^2 + w^2)^{\frac{1}{2}} \right| + Z_{vv}' v^2 \# \right]$$

$$+ \frac{\rho}{2} \ell^2 u^2 \left[Z_{\delta s}' \delta s + Z_{\delta b}' \delta b \right]$$

$$+ \frac{\rho}{2} \ell^2 \left[Z_{wn'}' (n' - 1) uw + Z_{w|w|n'}' (n' - 1) w \left| (v^2 + w^2)^{\frac{1}{2}} \right| \right]$$

$$+ \frac{\rho}{2} \ell^2 u^2 Z_{\delta s n'}' (n' - 1) \delta_s$$

$$+ \frac{\rho}{2} \ell^2 (F_z)_{vs} \frac{v^2 + w^2}{U} v \operatorname{Sin} \omega t$$

$$+ \Sigma W_i \cos \theta \cos \phi$$

\# Multiplied by $\frac{u}{U}$ for large angles of attack near -90°

Note 1
when not multiplied by $\frac{u}{U}$ add to Z_{vp}'

Equation of Motion About the Body Axis System x-Axis

$$I_x\dot{p} + (I_z - I_y)\,qr = \frac{\rho}{2}\ell^5 \left[K'_{\dot{p}}\,\dot{p} + K'_{qr}\,qr + K'_{\dot{r}}\,\dot{r} + K'_{p|p|}\,p|p|\right]$$

$$+ \frac{\rho}{2}\ell^4 \left[K'_{p}\,up + K'_{r}\,ur + K'_{\dot{v}}\,\dot{v} + K'_{wp}\,wp\right]$$

$$+ \frac{\rho}{2}\ell^3 \left[K'_{*}\,u^2 + K'_{v}\,uv + K'_{v|v|}\,v\left|(v^2 + w^2)^{\frac{1}{2}}\right|\right]$$

$$+ \frac{\rho}{2}\ell^3\, K'_{vw}\,vw$$

$$+ \frac{\rho}{2}\ell^3\, u^2 K'_{\delta r}\,\delta_r$$

$$+ Bz_B \sin\phi \cos\theta$$

Equation of Motion About the Body Axis System y-Axis

Note 1

$$I_y\dot{q} + (I_x - I_z)\, rp = \frac{\rho}{2}\ell^5 \left[M_{\dot{q}}{}'\,\dot{q} + M_{rr}{}'\, r^2\# + M_{rp}{}'\, rp + \Delta M_{rp}{}'\, rp\#\right]$$

$$+ \frac{\rho}{2}\ell^4 \left[M_{q}{}'\, uq + M_{|q|\delta s}{}'\, u|q|\delta s + M_{|w|q}{}'\, |(v^2 + w^2)^{\frac{1}{2}}|q\right]$$

$$+ \frac{\rho}{2}\ell^4 \left[M_{\dot{w}}{}'\,\dot{w} + M_{vr}{}'\, vr\# + M_{vp}{}'\, vp\#\right]$$

$$+ \frac{\rho}{2}\ell^4 M_{qn'}{}'\,(n' - 1)\, uq$$

$$+ \frac{\rho}{2}\ell^3 \left[M_{*}{}'\, u^2 + M_{w}{}'\, uw + M_{w|w|}{}'\, w|(v^2 + w^2)^{\frac{1}{2}}|\right]$$

$$+ \frac{\rho}{2}\ell^3 \left[M_{|w|}{}'\, u|w| + M_{ww}{}'\, |w(v^2 + w^2)^{\frac{1}{2}}| + M_{vv}{}'\, v^2\#\right]$$

$$+ \frac{\rho}{2}\ell^3\, u^2 \left[M_{\delta s}{}'\,\delta_s + M_{\delta b}{}'\,\delta_b\right]$$

$$+ \frac{\rho}{2}\ell^3 M_{wn'}{}'\,(n' - 1)\, uw$$

$$+ \frac{\rho}{2}\ell^3 M_{w|w|n'}{}'\,(n' - 1)\, w|(v^2 + w^2)^{\frac{1}{2}}|$$

$$+ \frac{\rho}{2}\ell^3\, u^2 M_{\delta sn'}{}'\,(n' - 1)\,\delta_s$$

$$+ Bz_B \sin\theta$$

$$- \Sigma W_i\, x_{ti} \cos\theta\cos\Phi$$

\# Multiply by $\frac{u}{U}$ for large angles of attack near $-90°$

Note 1
when not multiplied by $\frac{u}{U}$ add to $M_{rp}{}'$

Equation of Motion About the Body Axis System z-Axis

$$I_z \dot{r} + (I_y - I_x)\, pq = \frac{\rho}{2} \ell^5 \left[N_{\dot{r}}{}' \, \dot{r} + N_{pq}{}' \, pq + N_{\dot{p}}{}' \, \dot{p} \right]$$

$$+ \frac{\rho}{2} \ell^4 \left[N_{r}{}' \, ur + N_{|r|\delta r}{}' \, u|r|\delta r + N_{|v|r}{}' \, \left|(v^2 + w^2)^{\frac{1}{2}}\right| r \right]$$

$$+ \frac{\rho}{2} \ell^4 \left[N_{p}{}' \, up + N_{\dot{v}}{}' \, \dot{v} + N_{wp}{}' \, wp \right]$$

$$+ \frac{\rho}{2} \ell^4 \; N_{rn'}{}' \, (n' - 1)\, ur$$

$$+ \frac{\rho}{2} \ell^3 \left[N_{*}{}' \, u^2 + N_{v}{}' \, uv + N_{v|v|}{}' \, v\left|(v^2 + w^2)^{\frac{1}{2}}\right| \right]$$

$$+ \frac{\rho}{2} \ell^3 \; u^2 N_{\delta r}{}' \, \delta_r$$

$$+ \frac{\rho}{2} \ell^3 \; u^2 N_{\delta rn'}{}' \, (n' - 1)\, \delta_r$$

$$+ \frac{\rho}{2} \ell^3 \; N_{vn'}{}' \, (n' - 1)\, uv$$

$$+ \frac{\rho}{2} \ell^3 \; N_{v|v|n'}{}' \, (n' - 1)\, v\left|(v^2 + w^2)^{\frac{1}{2}}\right|$$

$$+ \frac{\rho}{2} \ell^3 \; N_{wv}{}' \; wv\#$$

$$+ \Sigma W_i \, x_{ti} \cos\theta \sin\phi$$

\# Multiply by $\frac{u}{U}$ for large angles of attack near -90°

AUXILIARY EQUATIONS

$$\dot{\phi} = p + \dot{\psi}\sin\theta$$

$$\dot{\theta} = (q - \dot{\psi}\cos\theta\sin\phi)/\cos\phi$$

$$\dot{\psi} = (r + \dot{\theta}\sin\phi)/\cos\theta\cos\phi$$

$$\dot{x}_o = u\cos\theta\cos\psi + v(\sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi) + w(\sin\phi\sin\psi + \cos\phi\sin\theta\cos\psi)$$

$$\dot{y}_o = u\cos\theta\sin\psi + v(\cos\phi\cos\psi + \sin\phi\sin\theta\sin\psi) + w(\cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi)$$

$$\dot{z}_o = -u\sin\theta + v\cos\theta\sin\phi + w\cos\theta\cos\phi$$

$$U = (u^2 + v^2 + w^2)^{\frac{1}{2}}$$

$$(F_x)_P = \frac{\rho}{2}\ell^2 u^2\left[a_1' + a_2' n' + a_3' n'^2\right] \quad \text{when} \quad k_1 < n'$$

$$= \frac{\rho}{2}\ell^2 u^2\left[b_1' + b_2' n' + b_3' n'^2\right] \quad \text{when} \quad k_2 < n' < k_1$$

$$= \frac{\rho}{2}\ell^2 u^2\left[c_1' + c_2' n' + c_3' n'^2\right] \quad \text{when} \quad k_3 < n' < k_2$$

$$= \frac{\rho}{2}\ell^2 u^2\left[d_1' + d_2' n' + d_3' n'^2\right] \quad \text{when} \quad n' < k_3$$

a_1', a_2', a_3'
b_1', b_2', b_3'
c_1', c_2', c_3'
d_1', d_2', d_3'

Sets of non-dimensional coefficients used in the propulsion equation above. The set which will be in effect at any time during a simulated maneuver will depend on the value of n' and the numbers k_1, k_2, k_3.

NOMENCLATURE

All symbols used in the equations of motion and in the auxiliary equations and relationships which appear in this report are defined below. Any dimensions involved will be consistent with the foot-pound-second system of units. All angles are in degrees. The Fortran variables corresponding to these symbols are shown in Appendix B .

SYMBOL	DEFINITION
•	A dot over any symbol signifies differentiation with respect to time.
B	Buoyancy force which is positive upwards.
m	Mass of the submarine including the water in the free flooding spaces.
ℓ	Overall length of the submarine
U	Linear velocity of origin of body axes relative to an earth-fixed axis system.
u	Component of U along the body x-axis.
v	Component of U along the body y-axis.
w	Component of U along the body z-axis.

u_c	Command speed: A steady value of u for a given propeller rpm when α, β and control surface angles are zero. Sign changes with propeller reversal.
x	Longitudinal axis of the body fixed coordinate axis system.
y	Transverse axis of the body fixed coordinate axis system.
z	Vertical axis of the body fixed coordinate axis system.
x_o	Distance along the x_o axis of an earth-fixed axis system.
y_o	Distance along the y_o axis of an earth-fixed axis system.
z_o	Distance along the z_o axis of an earth-fixed axis system.
p	Component of angular velocity about the body fixed x-axis.
q	Component of angular velocity about the body fixed y-axis.
r	Component of angular velocity about the body fixed z-axis.
z_B	The z coordinate of the center of buoyance (CB) of the submarine.

α	Angle of attack.
β	Angle of drift.
δb, D_b	Deflection of bowplane (or sailplane)
δr, D_r	Deflection of rudder.
δs, D_s	Deflection of sternplane.
n'	The ratio u_c/u.
θ	Pitch angle.
Ψ	Yaw angle.
ϕ	Roll angle.
ρ	Mass density of sea water.
W_i	Weight of water blown from a particular ballast tank identified by the integer assigned to the index i.
ω	Angular velocity.
t	Time.
x_{ti}	Location along the body x-axis of the center of mass of the i^{th} ballast tank when this tank is filled with sea water.

$(F_x)_p$	Propulsion force (see auxiliary equations and relationships).
I_x	Moment of inertia of a submarine about the x-axis.
I_y	Moment of inertia of a submarine about the y-axis.
I_z	Moment of inertia of a submarine about the z-axis.
$K_{\dot{p}}'$, K_p', $K_{p\|p\|}'$, K_{qr}', K_r', $K_{\dot{r}}'$, $K_{\dot{v}}'$, K_{wp}', K_*', K_v', $K_{v\|v\|}'$, K_{vw}', $K_{\delta r}'$	Non-dimensional constants each of which is assigned to a particular force term in the equation of motion about the body x-axis.
$M_{\dot{q}}'$, M_{rr}', M_{rp}', $\Delta M_{rp}'$, M_q', $M_{\|q\|\delta s}'$, $M_{\|w\|q}'$, $M_{\dot{w}}'$, M_{vr}', M_{vp}', M_{qn}', M_*', M_w', $M_{w\|w\|}'$, $M_{\|w\|}'$, M_{ww}', M_{vv}', $M_{\delta s}'$, $M_{\delta b}'$, M_{wn}', $M_{w\|w\|n}'$, $M_{\delta sn}'$	Non-dimensional constants each of which is assigned to a particular force term in the equation of motion about the body y-axis.

$N_{\dot{r}}'$, N_{pq}', $N_{\dot{p}}'$, N_r', $N_{|r|\delta r}'$, $N_{|v|r}'$,
N_p', $N_{\dot{v}}'$, N_{wp}', N_{rn}', N_*', N_v',
$N_{v|v|}'$, $N_{\delta r}'$, $N_{\delta rn}'$, N_{vn}', $N_{v|v|n}'$,
N_{wv}'

Non-dimensional constants each of which is assigned to a particular force term in the equation of motion about the body z-axis.

X_{qq}', X_{rr}', X_{rp}', $X_{\dot{u}}'$, X_{vr}', X_{wq}',
X_{uu}', X_{vv}', X_{ww}', $X_{\delta r\delta r}'$, $X_{\delta s\delta s}'$,
$X_{\delta b\delta b}'$, X_{vvn}', X_{wwn}', $X_{\delta s\delta sn}'$,
$X_{\delta r\delta rn}'$

Non-dimensional constants each of which is assigned to a particular force term in the equation of motion along the body x-axis.

$Y_{\dot{r}}'$, $Y_{\dot{p}}'$, Y_{pq}', $Y_{p|p|}$, $Y_{\dot{v}}'$, Y_{wp}',
$Y_{v|r|}'$, Y_r', $Y_{|r|\delta r}'$, Y_p', Y_{rn}',
Y_*', Y_v', $Y_{v|v|}'$, $Y_{\delta r}'$, $Y_{\delta rn}'$,
Y_{vn}', $Y_{v|v|n}'$, Y_{wv}', $(F_y)_{vs}$

Non-dimensional constants each of which is assigned to a particular force term in the equation of motion along the body y-axis

$Z_{\dot{q}}'$, Z_{rr}', Z_{rp}', $Z_{\dot{w}}'$, Z_{vr}, Z_{vp}',
$\Delta Z_{vp}'$, Z_q', $Z_{|q|\delta s}$, $Z_{w|q|}'$,
Z_{qn}', Z_*', Z_w', $Z_{w|w|}'$, $Z_{|w|}'$,
Z_{vw}', Z_{vv}', $Z_{\delta s}'$, $Z_{\delta b}'$, Z_{wn}',
$Z_{w|w|n}'$, $Z_{\delta sn}'$, $(F_z)_{vs}$

Non-dimensional constants each of which is assigned to a particular force term in the equation of motion along the body z-axis

```
// EXEC DSL
//DSL.INPUT DD *
*                        PROGRAM #1
TITLE SUBMARINE SIMJLATION VERTICAL PLANE OVERSHOOT
* A SUBMARINE IS SIMULATED IN 6 DEGREES OF FREEDOM. THE PARAMETERS
* ARE OBTAINED FROM THE NSRDC USERS GUIDE FOR SIMULATING SUBMARINE
* MOTION. THE VARIABLES AAA-AFF MUST BE IN THE ORDER LISTED TO
* USE FUNCTION VALUE. THE PLANES ARE COTROLLED BY SPECIFYING DSOD
* THE RATE OF MOVEMENT IS DETERMINED BY THE CONSTANT PLRT
* WHERE PLRT IS IN RADIANS PER SECOND.
* THE VERTICAL PLANE OVERSHOOT IS ACCOMPLISHED BY POSITIONING
* THE STERN PLANES INITIALLY TO TO +20 DEGREES  WHEN THE
* PITCH ANGLE REACHES -5.0 DEGREES THE STERN PLANE ANGLE IS REVERSED
PARAM LC=415.0,UC=25.33,ML=.0087445,A1=-.001,A2=-.00095,A3=.00195
PARAM IX=7.3114E-06,IY=5.6867E-04,IZ=5.6867E-04
PARAM XUDOT=-.00015,XVR=.011,XWQ=-.0075,XVV=.0065,XWW=.002,XDRDR=-.0028
PARAM XDSDS=-.0025,XDBDB=-.0026,XQQ=-.0002,XRR=-.00009,XRP=.00025
PARAM YVDOT=-.011,YWP=.0075,YV=-.021,Y1V1V=-.06,YR=.003,YV1R1=-.0073
PARAM YP=-.0007,YRDOT=.00009,YPDOT=-.0003,YDR=.0062,YPQ=.0002
PARAM YWV=-.065
PARAM ZWDOT=-.0075,ZVP=-.007,ZS=-.0001,ZW=-.011,ZW1W1=-.03,ZVV=.065
PARAM ZQ=-.0045,ZW1Q1=-.006,ZVR=-.008,ZRR=-.0015,ZDS=-.005,ZDB=-.0025
PARAM ZQDOT=-.0002,Z1W1=0.0,ZWW=0.0,ZRP=-.0009
PARAM KPDOT=-3.0E-06,KQR=-.0001,KRDOT=-7.0E-06,K1P1P=-8.0E-07,KV=-.0007
PARAM K1V1V=-.0009,KP=-3.5E-05,KR=-4.0E-05,KVDOT=-.00025,KVW=.0035
PARAM KDR=7.0E-05,KWP=2.5E-04
PARAM MQDOT=-.0004,MRP=.00015,MS=4.0E-05,MW=.003,M1W1W=-.005,MVV=.015
PARAM MQ=-.0025,M1W1Q=-.002,MVR=-.004,MRR=-.00055,MWDOT=-.0002
PARAM MDS=-.0025,MDB=.0005,M1W1=0.0,MVP=.0009
PARAM NRDOT=-5.0E-04,NPQ=-4.0E-04,NPDOT=-7.0E-06
PARAM NV=-.0075,N1V1V=.014
PARAM NR=-.003,N1V1R=-.0045,NP=-2.0E-06,NVDOT=.0003,NDR=-.003
PARAM NWV=.015,NWP=-.0002
PARAM BZB=1.011413E-03
PARAM PLRT=.1222
INCON YADOT=0.0,RODOT=0.0,PIDOT=0.0
INCON  DS=0.0,DB=0.0,DR=0.0
INTGER NPLOT
CONST NPLOT=1
CONTRL FINTIM=250.0,DELT=.01,DELS=.5
INITIAL
      LC2=LC**2
      IZX=IZ-IX
      IYX=IY-IX
      IZY=IZ-IY
      AAA=ML-XUDOT
      AAB=0.0
      AAC=0.0
```

```
AAD=0.0
AAE=0.0
AAF=0.0
ABA=0.0
ABB=ML-YVDOT
ABC=0.0
ABD=-KVDOT/LC
ABE=0.0
ABF=-NVDOT/LC
ACA=0.0
ACB=0.0
ACC=ML-ZWDOT
ACD=0.0
ACE=-MWDOT/LC
ACF=0.0
ADA=0.0
ADB=-YPDOT*LC
ADC=0.0
ADD=IX-KPDOT
ADE=0.0
ADF=-NPDOT
AEA=0.0
AEB=0.0
AEC=-ZQDOT*LC
AED=0.0
AEE=IY-MQDOT
AEF=0.0
AFA=0.0
AFB=-YRDOT*LC
AFC=0.0
AFD=-KRDOT
AFE=0.0
AFF=IZ-NRDOT
DEL=VALUE(AAA,0,0)
COFAA=VALUE(AAA,1,1)
COFAB=VALUE(AAA,2,1)
COFAC=VALUE(AAA,3,1)
COFAD=VALUE(AAA,4,1)
COFAE=VALUE(AAA,5,1)
COFAF=VALUE(AAA,6,1)
COFBA=VALUE(AAA,1,2)
COFBB=VALUE(AAA,2,2)
COFBC=VALUE(AAA,3,2)
COFBD=VALUE(AAA,4,2)
COFBE=VALUE(AAA,5,2)
COFBF=VALUE(AAA,6,2)
COFCA=VALUE(AAA,1,3)
COFCB=VALUE(AAA,2,3)
```

```
      COFCC=VALUE(AAA,3,3)
      COFCD=VALUE(AAA,4,3)
      COFCE=VALUE(AAA,5,3)
      COFCF=VALUE(AAA,6,3)
      COFDA=VALUE(AAA,1,4)
      COFDB=VALUE(AAA,2,4)
      CCFDC=VALUE(AAA,3,4)
      COFDD=VALUE(AAA,4,4)
      COFDE=VALUE(AAA,5,4)
      COFDF=VALUE(AAA,6,4)
      COFEA=VALUE(AAA,1,5)
      COFEB=VALUE(AAA,2,5)
      COFEC=VALUE(AAA,3,5)
      COFED=VALUE(AAA,4,5)
      COFEE=VALUE(AAA,5,5)
      COFEF=VALUE(AAA,6,5)
      COFFA=VALUE(AAA,1,6)
      COFFB=VALUE(AAA,2,6)
      COFFC=VALUE(AAA,3,6)
      COFFD=VALUE(AAA,4,6)
      COFFE=VALUE(AAA,5,6)
      COFFF=VALUE(AAA,6,6)
DERIVATIVE
NOSORT
      DEPTH=INTGRL(0.0,ZODOT)
      U=INTGRL(25.33,UDOT)
      V=INTGRL(0.,VDOT)
      W=INTGRL(0.,WDOT)
      P=INTGRL(0.,PDOT)
      Q=INTGRL(0.,QDOT)
      R=INTGRL(0.,RDOT)
      ROLL=INTGRL(0.0,RODOT)
      PITCH=INTGRL(0.0,PIDOT)
      YAW=INTGRL(0.0,YADOT)
* PLANE ANGLE GENERATOR
      DSER=DSOD-DS
      IF(DSER.EQ.0.0) GOTO 22
      IF(DSER.LT.0.0) GOTO 27
      IF(DSTMP.EQ.1) GOTO 28
      DSTMM=0
      DSTMP=1
      DSC=DS
      DST=TIME
   28 DSA=(TIME-DST)*PLRT
      DS=DSC+DSA
      GOTO 21
   27 IF(DSTMM.EQ.1) GOTO 29
      DSTMP=0
```

```
      DSTMM=1
      DSC=DS
      DST=TIME
   29 DSA=(DST-TIME)*PLRT
      DS=DSC+DSA
      GOTO 21
   22 DSTMM=0
      DSTMP=0
   21 CONTINUE
* AUXILIARY EQUATIONS
      ZODOT=-U*SIN(PITCH)+V*COS(PITCH)*SIN(ROLL)...
      +W*COS(PITCH)*COS(ROLL)
      PIDOT=Q*COS(ROLL)-R*SIN(ROLL)
      YADOT=(R*COS(ROLL)+Q*SIN(ROLL))/COS(PITCH)
      RODOT=P+YADOT*SIN(PITCH)
      DSGRA=DS*57.296
      DBGRA=DB*57.296
      PITGRA=180.*PITCH/3.14159
      PA1=XDRDR*U*U*DR*DR/LC
      PA2=XDSDS*U*U*DS*DS/LC
      PA3=XDBDB*U*U*DB*DB/LC
      PB1=YDR*U*U*DR/LC
      PC2=ZDS*U*U*DS/LC
      PC3=ZDB*U*U*DB/LC
      PD1=KDR*U*U*DR/LC2
      PE2=MDS*U*U*DS/LC2
      PE3=MDB*U*U*DB/LC2
      PF1=NDR*U*U*DR/LC2
      PA=PA1+PA2+PA3
      PB=PB1
      PC=PC2+PC3
      PD=PD1
      PE=PE2+PE3
      PF=PF1
*NON LINEAR RELATIONS
      ABV=ABS(V)
      ABW=ABS(W)
      ABP=ABS(P)
      ABQ=ABS(Q)
      ABR=ABS(R)
      VVWW=V*V+W*W
      AVW=SQRT(VVWW)
      ABWP=FCNSW(W,-1.0,0.0,1.0)
      ABVP=FCNSW(V,-1.0,0.0,1.0)
      SA1=+LC*(XQQ*Q**2+XRR*R**2+XRP*R*P)
      SA2=+(ML*V*R+XVR*V*R+XWQ*W*Q-ML*W*Q)
      SA3=+(XVV*V**2+XWW*W**2)/LC-SIN(PITCH)*(AT+FT+AU)
      SA4=+(A1*U**2+A2*U*UC+A3*UC**2)/LC
```

```
      SB1=+LC*YPQ*P*Q
      SB2=+(YWP*W*P+YV1R1*ABR*AVW*ABVP+ML*W*P-ML*U*R)
      SB3=+(YWV*W*V+Y1V1V*AVW*V)/LC+SIN(ROLL)*COS(PITCH)...
      *(AT+FT+AU)
      SB4=(YR*R+YP*P+YV*V/LC)*U
      SC1=LC*R*(ZRR*R+ZRP*P)
      SC2=+(ZVP*V*P+ZVR*V*R+ZW1Q1*ABQ*AVW*ABWP+ML*U*Q-ML*P*V)
      SC3=+(ZWW*W**2+ZVV*V**2+ZW1W1*W*AVW+U*Z1W1*ABW+U*U*ZS)/LC
      SC4=ZQ*U*Q+ZW*U*W/LC+COS(PITCH)*COS(ROLL)*(AT+FT+AU)
      SD1=+(KQR*Q*R+K1P1P*ABP*P)-IZY*Q*R
      SD2=(KWP*W*P-BZB*SIN(ROLL)*COS(PITCH))/LC
      SD3=+(K1V1V*V*AVW+KVW*V*W+KS*U**2)/LC2
      SD4=((KP*P+KR*R)/LC+KV*V/LC2)*U
      SE1=(MRP*P+MRR*R+IZX*P)*R
      SE2=((MVR*R+MVP*P)*V+M1W1Q*AVW*Q-BZB*SIN(PITCH))/LC
      SE3=(MVV*V**2+MWW*W**2+M1W1W*AVW*W+M1W1*U*AVW+U**2*MS)/LC2
      SE4=MQ*U*Q/LC+(MW*U*W-(175.5*FT-219.5*AT)*COS(PITCH)...
      *COS(ROLL))/LC2
      SF1=(NPQ-IYX)*P*Q
      SF2=+(NWP*W*P+N1V1R*AVW*R)/LC
      SF3=(NWV*W+N1V1V*AVW)*V/LC2
      SF4=(NP*P+NR*R)*U/LC+(NV*U*V+(175.5*FT-219.5*AT)*...
      COS(PITCH)*SIN(ROLL))/LC2
      SA=SA1+SA2+SA3+SA4
      SB=SB1+SB2+SB3+SB4
      SC=SC1+SC2+SC3+SC4
      SD=SD1+SD2+SD3+SD4
      SE=SE1+SE2+SE3+SE4
      SF=SF1+SF2+SF3+SF4
      ZA=SA+PA
      ZB=SB+PB
      ZC=SC+PC
      ZD=SD+PD
      ZE=SE+PE
      ZF=SF+PF
* EQUATIONS OF MOTION
      UDOT=(COFAA*ZA+COFAB*ZB+COFAC*ZC+COFAD*ZD+COFAE*ZE+COFAF*ZF)/DEL
      VDOT=(COFBA*ZA+COFBB*ZB+COFBC*ZC+COFBD*ZD+COFBE*ZE+COFBF*ZF)/DEL
      WDOT=(COFCA*ZA+COFCB*ZB+COFCC*ZC+COFCD*ZD+COFCE*ZE+COFCF*ZF)/DEL
      PDOT=(COFDA*ZA+COFDB*ZB+COFDC*ZC+COFDD*ZD+COFDE*ZE+COFDF*ZF)/DEL
      QDOT=(COFEA*ZA+COFEB*ZB+COFEC*ZC+COFED*ZD+COFEE*ZE+COFEF*ZF)/DEL
      RDOT=(COFFA*ZA+COFFB*ZB+COFFC*ZC+COFFD*ZD+COFFE*ZE+COFFF*ZF)/DEL
SORT
SAMPLE
      CALL DRWG(1,1,TIME,DEPTH)
      CALL DRWG(2,1,TIME,DSGRA)
      CALL DRWG(2,2,TIME,PITGRA)
TERMINAL
```

```
      CALL ENDRW(NPLOT)
END
STOP
FORTRAN
      FUNCTION VALUE(B,I,MM)
COMMON
      DIMENSION A(6,6),B(6,6)
      DO 1 M1=1,6
      DO 1 M2=1,6
    1  A(M1,M2)=B(M1,M2)
      IF(I.EQ.0) GO TO 100
      A(I,1)=0.
      A(I,2)=0.
      A(I,3)=0.
      A(I,4)=0.
      A(I,5)=0.
      A(I,6)=0.
      A(1,MM)=0.
      A(2,MM)=0.
      A(3,MM)=0.
      A(4,MM)=0.
      A(5,MM)=0.
      A(6,MM)=0.
      A(I,MM)=1.
  100 CONTINUE
      WRITE(6,49) I,MM
   49 FORMAT(/,' ','DETERMINANT FOR COF',I1,I1,':')
      DO 50 M1=1,6
      WRITE(6,51)(A(M1,M2),M2=1,6)
   51 FORMAT(10(1X,E13.6))
   50 CONTINUE
      NN=6
      DD=1.D0
   10 DO 34 L=1,NN
       KP=0
      C=0.0
      DO 12 KKK=L,NN
      IF(C-ABS(A(KKK,L)))11,12,12
   11 C=ABS(A(KKK,L))
      KZ=KKK
   12  CONTINUE
       IF(L-KZ)13,20,20
   13 DO 14 J=L,NN
       C=A(L,J)
       A(L,J)=A(KZ,J)
   14  A(KZ,J)=C
      DD=-DD
   20 IF(L-NN)31,40,40
```

```
31   LP1=L+1
     DO 34 KKK=LP1,NN
     IF(A(KKK,L))32,34,32
32   RATIO=A(KKK,L)/A(L,L)
     DO 33 J=LP1,NN
33   A(KKK,J)=A(KKK,J)-RATIO*A(L,J)
34    CONTINUE
40   DO 41 KKK=1,NN
41   DD=DD*A(KKK,KKK)
     D=DD
     VALUE=D
     WRITE(6,52) I,MM,VALUE
52   FORMAT(' ','COF',I1,I1,'=',E15.6)
     RETURN
     END
//PLOT.SYSIN DD *
DRUREY  SUBMARINE VERTICAL PLANE OVERSHOOT
DEPTH VS TIME                                                    4
                                  5.0       7.0
DRUREY  SUBMARINE VERTICAL PLANE OVERSHOOT
PITCH AND DS VS TIME                                             4
                                  5.0       7.0
```

```
// EXEC DSL
//DSL.INPUT DD *
*                                   PROGRAM #2
* OPTIMAL GAINS ARE CALCULATED UNTIL THE TIME REQUIREMENTS OF AN IF
* STATEMENT ARE  MET. THE TIME REQUIREMENT IS SELECTED TO ENSURE A
* STEADY STATE CONDITION IS REACHED. THE STEADY STATE GAINS ARE USED
* IN THE DEPTH CONTROLLER. THE SIMULATION DOES NOT START UNTIL THE
* STEADY STATE IS ACHIEVED.
*     THE WEIGHTING FACTOR IN THE GAIN EQUATION FOR DEPTH ERROR IS THE
* PARAMETER E. THE WEIGHT FOR PITCH ERROR IS D. THE CONTROL INPUT IS
* WEIGHTED WITH C.
* PARAMETERS ARE ASSIGNED AS FOLLOWS
* MAX PLANE RATE---PLRT
* PUMP RATE---KPR
* INITIAL TRIM TANK LEVELS---AT,FT,AU
* DEADZONE IN THE VERTICAL VELOCITY CHANNEL---DZ1
* DEADZONE IN THE ROTATIONAL VELOCITY CHANNEL---DY1
* THE PITCH ERROR AND DEPTH ERROR ARE LIMITED IN THE PROGRAM.
* THE PUMP IS OPERATED AS FOLLOWS
* START PUMP---PUMP=1
* PUMP BETWEEN FT AND AT---FTAT=1,AUSE=0
* PUMP AUX---AUSE=1,FTAT=0
* MAXIMUM AMOUNT TO BE PUMPED---LIMID=WEIGHT/415**3
* DIRECTION---DIRCT=1.0 PUMPS FROM FT TO AT OR FROM SEA TO AUX
* TO REVERSE DIRECTION OF PUMP---DIRCT=-1.0
PARAM C=10.0,D=3000.0,E=1.0
PARAM DEL=.18901E-16,COFAA=.212502E-14,COFAB=0.0,COFAC=0.0
PARAM COFAD=0.0,COFAE=0.0,COFAF=0.0,COFBA=0.0
PARAM COFBB=.153152E-14,COFBC=0.0,COFBD=-.186106E-10
PARAM COFBE=0.0,COFBF=.17543E-12,COFCA=0.0,COFCB=0.0
PARAM COFCC=.11665E-14,COFCD=0.0,COFCE=-.999506E-13
PARAM COFCF=0.0,COFDA=0.0,COFDB=-.905797E-16,COFDC=0.0
PARAM COFDD=.294191E-11,COFDE=0.0,COFDF=-.224359E-13
PARAM COFEA=0.0,COFEB=0.0,COFEC=-.58035E-18,COFED=0.0
PARAM COFEE=.19562E-13,COFEF=0.0,COFFA=0.0,COFFB=.162929E-17
PARAM COFFC=0.0,COFFD=-.318591E-13,COFFE=0.0,COFFF=.179521E-13
PARAM LC=415.0,UC=25.33,ML=.0087445,A1=-.001,A2=-.00095,A3=.00195
PARAM IX=7.3114E-06,IY=5.6867E-04,IZ=5.6867E-04
PARAM XUDOT=-.00015,XVR=.011,XWQ=-.0075,XVV=.0065,XWW=.002,XDRDR=-.0028
PARAM XDSDS=-.0025,XDBDB=-.0026,XQQ=-.0002,XRR=-.00009,XRP=.00025
PARAM YVDOT=-.011,YWP=.0075,YV=-.021,Y1V1V=-.06,YR=.003,YV1R1=-.0073
PARAM YP=-.0007,YRDOT=.00009,YPDOT=-.0003,YDR=.0062,YPQ=.0002
PARAM YWV=-.065
PARAM ZWDOT=-.0075,ZVP=-.007,ZS=-.0001,ZW=-.011,ZW1W1=-.03,ZVV=.065
PARAM ZQ=-.0045,ZW1Q1=-.006,ZVR=-.008,ZRR=-.0015,ZDS=-.005,ZDB=-.0025
PARAM ZQDOT=-.0002,Z1W1=0.0,ZWW=0.0,ZRP=-.0009
PARAM KPDOT=-3.0E-06,KQR=-.0001,KRDOT=-7.0E-06,K1P1P=-8.0E-07,KV=-.0007
PARAM K1V1V=-.0009,KP=-3.5E-05,KR=-4.0E-05,KVDOT=-.00025,KVW=.0035
```

```
PARAM KDR=7.0E-05,KWP=2.5E-04
PARAM MQDOT=-.0004,MRP=.00015,MS=4.0E-05,MW=.003,M1W1W=-.005,MVV=.015
PARAM MQ=-.0025,M1W1Q=-.002,MVR=-.004,MRR=-.00055,MWDOT=-.0002
PARAM MDS=-.0025,MDB=.0005,M1W1=0.0,MVP=.0009
PARAM NRDOT=-5.0E-04,NPQ=-4.0E-04,NPDOT=-7.0E-06
PARAM NV=-.0075,N1V1V=.014
PARAM NR=-.003,N1V1R=-.0045,NP=-2.0E-06,NVDOT=.0003,NDR=-.003
PARAM NWV=.015,NWP=-.0002
PARAM EZB=1.011413E-03
PARAM PLRT=.1222
PARAM KPR=1.8E-06
INCON YADOT=0.0,RODOT=0.0,PIDOT=0.0
INCON FT=-2.0E-05,AT=2.0E-05,AU=4.0E-04
INCON  DS=0.0,DB=0.0,DR=0.0
PARAM DZ1=.04
PARAM DY1=.0008
INTGER PUMP,FTAT,AUSE
INTGER PUPLS,PUMIN
INTGER CBTMP,CBTMM,DSTMP,DSTMM
INTGER PUFTA,PUAUX
INTGER NPLOT
CONST NPLOT=1
CONTRL FINTIM=350.0,DELT=.01,DELS=.5
INITIAL
      CI=1.0/C
      UC2=UC**2
      LC2=LC**2
      LC3=LC2*LC
      IZX=IZ-IX
      IYX=IY-IX
      IZY=IZ-IY
      ACC=ML-ZWDOT
      ACE=MWDOT/LC
      AEC=ZQDOT*LC
      AEE=IY-MQDOT
      INV=1.0/(ACC*AEE-ZQDOT*MWDOT)
      A11=(AEE*ZW+AEC*MW/LC)*INV*UC/LC
      A12=(ACE*ZW+ACC*MW/LC)*INV*UC/LC
      A21=(AEE*ZQ+AEC*MQ/LC)*INV*UC
      A22=(ACE*ZQ+ACC*MQ/LC)*INV*UC
      B11=(AEE*ZDS+AEC*MDS/LC)*INV*UC2/LC
      B12=(ACE*ZDS+ACC*MDS/LC)*INV*UC2/LC
      B21=(AEE*ZDB+AEC*MDB/LC)*INV*UC2/LC
      B22=(ACE*ZDB+ACC*MDB/LC)*INV*UC2/LC
      B1=A11/UC
      B2=A21/UC
      B3=A12/UC
      B4=A22/UC
```

```
      B5=B11/UC2
      B6=B21/UC2
      B7=B12/UC2
      B8=B22/UC2
      DY2=-DY1
      DZ2=-DZ1
DERIVATIVE
NOSORT
      DEPTH=INTGRL(0.0,ZODOT)
      U=INTGRL(25.33,UDOT)
      V=INTGRL(0.,VDOT)
      W=INTGRL(0.,WDOT)
      P=INTGRL(0.,PDOT)
      Q=INTGRL(0.,QDOT)
      R=INTGRL(0.,RDOT)
      ROLL=INTGRL(0.0,RODOT)
      PITCH=INTGRL(0.0,PIDOT)
      YAW=INTGRL(0.0,YADOT)
      Q1=INTGRL(0.0,QD1)
      W1=INTGRL(0.0,WD1)
      K11=INTGRL(0.0,KD11)
      K12=INTGRL(0.0,KD12)
      K13=INTGRL(0.0,KD13)
      K14=INTGRL(0.0,KD14)
      K22=INTGRL(0.0,KD22)
      K23=INTGRL(0.0,KD23)
      K24=INTGRL(0.0,KD24)
      K33=INTGRL(0.0,KD33)
      K34=INTGRL(0.0,KD34)
      K44=INTGRL(0.0,KD44)
      F11=B11*K12+B12*K14
      F12=B21*K12+B22*K14
      F21=B11*K22+B12*K24
      F22=B21*K22+B22*K24
      F31=B11*K23+B12*K34
      F32=B21*K23+B22*K34
      F41=B11*K24+B12*K44
      F42=B21*K24+B22*K44
      KD11=E-CI*(F11**2+F12**2)
      KD12=K11+K12*A11+K14*A12-CI*(F21*F11+F22*F12)
      KD13=-CI*(F31*F11+F32*F12)
      KD14=K12*A12+K13+K14*A22-CI*(F41*F11+F42*F12)
      KD22=2.0*K12+2.0*K22*A11+2.0*K24*A12-CI*(F21**2+F22**2)+A
      KD23=K13+K23*A11+K34*A12-CI*(F31*F21+F32*F22)
      KD24=K14+K23*A11+K44*A12+K22*A21+K23+K24*A22-CI*...
      (F41*F21+F42*F22)
      KD33=-CI*(F31**2+F32**2)+D
      KD34=K23*A21+K33+K34*A22-CI*(F41*F31+F42*F32)
```

```
      KD44=2.0*K24*A21+2.0*K34+2.0*K44*A22-CI*(F41**2+F42**2)+B
      IF(TIME.GE.20.0) GOTO 30
      ZODOT=0.0
      UDOT=0.0
      VDOT=0.0
      WDOT=0.0
      PDOT=0.0
      QDOT=0.0
      RDOT=0.0
      RODOT=0.0
      PIDOT=0.0
      YADOT=0.0
      GOTO 31
   30 CONTINUE
      ZM1=K12*641.609
      ZM2=K14*641.609
      ZM3=K22*16251.953
      ZM4=K23*641.609
      ZM5=K24*16251.953
      ZM6=K34*641.609
      ZM7=K44*16251.953
      Y11=-(B5*ZM1+B7*ZM2)*CI
      Y12=-(B6*ZM1+B8*ZM2)*CI
      Y21=-(B5*ZM3+B7*ZM5)*CI
      Y22=-(B6*ZM3+B8*ZM5)*CI
      Y31=-(B5*ZM4+B7*ZM6)*CI
      Y32=-(B6*ZM4+B8*ZM6)*CI
      Y41=-(B5*ZM5+B7*ZM7)*CI
      Y42=-(B6*ZM5+B8*ZM7)*CI
*  AUTOMATIC CONTROLLER FOR THE FAIRWATER AND STERN PLANES
      PERR=PITCH-PORD
      ZOER=DEPTH-ZODR
      IF(PERR.LT.-.174) PERR=-.174
      IF(PERR.GT..174) PERR=.174
      IF(ZOER.LT.-2.0) ZOER=-2.0
      IF(ZOER.GT.2.0) ZOER=2.0
      DSAD=Y11*ZOER+Y21*ZODOT/U+Y31*PERR+Y41*PIDOT/U
      DBAD=Y12*ZOER+Y22*ZODOT/U+Y32*PERR+Y42*PIDOT/U
      DSOD=REALPL(0.0,.1,DSAD)
      DBOD=REALPL(0.0,.1,DBAD)
*  PLANE ANGLE GENERATOR
      DSER=DSOD-DS
      DBER=DBOD-DB
      IF(DBER.EQ.0.0) GOTO 12
      IF(DBER.LT.0.0) GOTO 17
      IF(DBIMP.EQ.1) GOTO 18
      DBTMM=0
      DBTMP=1
```

```
      CBC=DB
      DBT=TIME
   18 DBA=(TIME-DBT)*PLRT
      DB=DBC+DBA
      GOTO 11
   17 IF(DBTMM.EQ.1) GOTO 19
      DBTMP=0
      DBTMM=1
      DBC=DB
      DBT=TIME
   19 DBA=(DBT-TIME)*PLRT
      DB=DBC+DBA
      GOTO 11
   12 DBTMM=0
      DBTMP=0
   11 CONTINUE
      IF(DSER.EQ.0.0) GOTO 22
      IF(DSER.LT.0.0) GOTO 27
      IF(DSTMP.EQ.1) GOTO 28
      DSTMM=0
      DSTMP=1
      DSC=DS
      DST=TIME
   28 DSA=(TIME-DST)*PLRT
      DS=DSC+DSA
      GOTO 21
   27 IF(DSTMM.EQ.1) GOTO 29
      DSTMP=0
      DSTMM=1
      DSC=DS
      DST=TIME
   29 DSA=(DST-TIME)*PLRT
      DS=DSC+DSA
      GOTO 21
   22 DSTMM=0
      DSTMP=0
   21 CONTINUE
* PUMP EQUATIONS
      IF(PUMP.EQ.1) GOTO 1
      GOTO 9
    1 IF(DIRCT.EQ.-1.0) GOTO 3
      IF(PUPLS.EQ.1) GOTO 2
      FWT=FT
      AFT=AT
      ALX=AU
      WW=TIME
      PUPLS=1
      PUMIN=0
```

```
      COUP=COUP+1.0
    2 MAXI=(TIME-WW)*KPR
      IF(MAXI.GT.LIMID) GOTO 6
      IF(FTAT.NE.1) GOTO 8
      GOTO 7
    3 IF(PUMIN.EQ.1) GOTO 4
      COUDN=COUDN+1.0
      FWT=FT
      AFT=AT
      AUX=AU
      WW=TIME
      PUPLS=0
      PUMIN=1
    4 MAXI=(WW-TIME)*KPR
      MAXIM=-MAXI
      IF(MAXIM.GE.LIMID) GOTO 6
      IF(FTAT.NE.1) GOTO 8
    7 FT=FWT-MAXI
      AT=AFT+MAXI
      GOTO 9
    8 IF(AUSE.NE.1) GOTO 6
      AU=AUX+MAXI
      GOTO 9
    6 PUMP=0
      FTAT=0
      AUSE=0
      PUPLS=0
      PUMIN=0
    9 CONTINUE
*  AUXILIARY EQUATIONS
      ZODOT=-U*SIN(PITCH)+V*COS(PITCH)*SIN(ROLL)...
      +W*COS(PITCH)*COS(ROLL)
      PIDOT=Q*COS(ROLL)-R*SIN(ROLL)
      YADOT=(R*COS(ROLL)+Q*SIN(ROLL))/COS(PITCH)
      RODOT=P+YADOT*SIN(PITCH)
      DSGRA=DS*57.296
      DBGRA=DB*57.296
      PITGRA=180.*PITCH/3.14159
      ROLGRA=ROLL*57.296
      PA1=XDRDR*U*U*DR*DR/LC
      PA2=XDSDS*U*U*DS*DS/LC
      PA3=XDBDB*U*U*DB*DB/LC
      PB1=YDR*U*U*DR/LC
      PC2=ZDS*U*U*DS/LC
      PC3=ZDB*U*U*DB/LC
      PD1=KDR*U*U*DR/LC2
      PE2=MDS*U*U*DS/LC2
      PE3=MDB*U*U*DB/LC2
```

```
      PF1=NDR*U*U*DR/LC2
      PA=PA1+PA2+PA3
      PB=PB1
      PC=PC2+PC3
      PD=PD1
      PE=PE2+PE3
      PF=PF1
*NON LINEAR RELATIONS
      ABV=ABS(V)
      ABW=ABS(W)
      ABP=ABS(P)
      ABQ=ABS(Q)
      ABR=ABS(R)
      VVWW=V*V+W*W
      AVW=SQRT(VVWW)
      ABWP=FCNSW(W,-1.0,0.0,1.0)
      ABVP=FCNSW(V,-1.0,0.0,1.0)
      SA1=+LC*(XQQ*Q**2+XRR*R**2+XRP*R*P)
      SA2=+(ML*V*R+XVR*V*R+XWQ*W*Q-ML*W*Q)
      SA3=+(XVV*V**2+XWW*W**2)/LC-SIN(PITCH)*(AT+FT+AU)
      SA4=+(A1*U**2+A2*U*UC+A3*UC**2)/LC
      SB1=+LC*YPQ*P*Q
      SB2=+(YWP*W*P+YV1R1*ABR*AVW*ABVP+ML*W*P-ML*U*R)
      SB3=+(YWV*W*V+Y1V1V*AVW*V)/LC+SIN(ROLL)*COS(PITCH)...
      *(AT+FT+AU)
      SB4=(YR*R+YP*P+YV*V/LC)*U
      SC1=LC*R*(ZRR*R+ZRP*P)
      SC2=+(ZVP*V*P+ZVR*V*R+ZW1Q1*ABQ*AVW*ABWP+ML*U*Q-ML*P*V)
      SC3=+(ZWW*W**2+ZVV*V**2+ZW1W1*W*AVW+U*Z1W1*ABW+U*U*ZS)/LC
      SC4=ZQ*U*Q+ZW*U*W/LC+COS(PITCH)*COS(ROLL)*(AT+FT+AU)
      SD1=+(KQR*Q*R+K1P1P*ABP*P)-IZY*Q*R
      SD2=(KWP*W*P-BZB*SIN(ROLL)*COS(PITCH))/LC
      SD3=+(K1V1V*V*AVW+KVW*V*W+KS*U**2)/LC2
      SD4=((KP*P+KR*R)/LC+KV*V/LC2)*U
      SE1=(MRP*P+MRR*R+IZX*P)*R
      SE2=((MVR*R+MVP*P)*V+M1W1Q*AVW*Q-BZB*SIN(PITCH))/LC
      SE3=(MVV*V**2+MWW*W**2+M1W1W*AVW*W+M1W1*U*AVW+U**2*MS)/LC2
      SE4=MQ*U*Q/LC+(MW*U*W-(175.5*FT-219.5*AT)*COS(PITCH)...
      *COS(ROLL))/LC2
      SF1=(NPQ-IYX)*P*Q
      SF2=+(NWP*W*P+N1V1R*AVW*R)/LC
      SF3=(NWV*W+N1V1V*AVW)*V/LC2
      SF4=(NP*P+NR*R)*U/LC+(NV*U*V+(175.5*FT-219.5*AT)*...
      COS(PITCH)*SIN(ROLL))/LC2
      SA=SA1+SA2+SA3+SA4
      SB=SB1+SB2+SB3+SB4
      SC=SC1+SC2+SC3+SC4
      SD=SD1+SD2+SD3+SD4
```

```
      SE=SE1+SE2+SE3+SE4
      SF=SF1+SF2+SF3+SF4
      ZA=SA+PA
      ZB=SB+PB
      ZC=SC+PC
      ZD=SD+PD
      ZE=SE+PE
      ZF=SF+PF
* EQUATIONS OF MOTION
      UDOT=(COFAA*ZA+COFAB*ZB+COFAC*ZC+COFAD*ZD+COFAE*ZE+COFAF*ZF)/DEL
      VDOT=(COFBA*ZA+COFBB*ZB+COFBC*ZC+COFBD*ZD+COFBE*ZE+COFBF*ZF)/DEL
      WDOT=(COFCA*ZA+COFCB*ZB+COFCC*ZC+COFCD*ZD+COFCE*ZE+COFCF*ZF)/DEL
      PDOT=(COFDA*ZA+COFDB*ZB+COFDC*ZC+COFDD*ZD+COFDE*ZE+COFDF*ZF)/DEL
      QDOT=(COFEA*ZA+COFEB*ZB+COFEC*ZC+COFED*ZD+COFEE*ZE+COFEF*ZF)/DEL
      RDOT=(COFFA*ZA+COFFB*ZB+COFFC*ZC+COFFD*ZD+COFFE*ZE+COFFF*ZF)/DEL
      HEVY=HEAVY/LC3
      WD1=(B1*W1+B2*Q1)*U+(B5*DS+B6*DB)*U**2+HEVY
      QD1=(B3*W1+B4*Q1)*U+(B7*DS+B8*DB)*U**2
      VERT=W-W1
      PRET=Q-Q1
      PRER=REALPL(0.0,30.0,PRET)
      VERR=REALPL(0.0,30.0,VERT)
      PUMPR=DEADSP(DZ2,DZ1,VERR)
      IF(PUMPR.EQ.0.0) GOTO 40
      IF(PUAUX.EQ.1) GOTO 42
      PUFTA=0
      PUAUX=1
      PUPLS=0
      PUMIN=0
   42 APUPR=ABS(PUMPR)
      PUMP=1
      LIMID=1.0
      AUSE=1
      FTAT=0
      DIRCT=-PUMPR/APUPR
      GOTO 51
   40 PUAUX=0
      PUMP=0
   41 CONTINUE
      PUMPS=DEADSP(DY2,DY1,PRER)
      IF(PUMPS.EQ.0.0) GOTO 50
      IF(PUFTA.EQ.1) GOTO 52
      PUFTA=1
      PUPLS=0
      PUMIN=0
   52 APUPS=ABS(PUMPS)
      APUPS=ABS(PUMPS)
      PUMP=1
```

```
      LIMID=1.0
      FTAT=1
      AUSE=0
      DIRCT=-PUMPS/APUPS
      GOTO 51
   50 LIMID=0.0
   51 CONTINUE
      ATT=AT*LC3
      AUT=AU*LC3
      FTT=FT*LC3
   31 CONTINUE
SAMPLE
      CALL DRWG(1,1,TIME,DSGRA)
      CALL DRWG(2,1,TIME,AUT)
      CALL DRWG(2,2,TIME,FTT)
      CALL DRWG(2,3,TIME,ATT)
      CALL DRWG(3,1,TIME,DBGRA)
      CALL DRWG(4,1,TIME,DEPTH)
      CALL DRWG(5,1,TIME,PITCH)
TERMINAL
      CALL ENDRW(NPLOT)
END
STOP
//PLCT.SYSIN DD *
DRUREY SUBMARINE TRIM CONTROL
STERN PLANE ANGLE VS TIME                                   4
                                   5.0        7.0
DRUREY SUBMARINE TRIM CONTROL
TANK LEVELS                                                 7
-10000.0  7500.0                   5.0        7.0
DRUREY SUBMARINE TRIM CONTROL
FAIRWATER PLANE ANGLE VS TIME                               4
                                   5.0        7.0
DRUREY SUBMARINE TRIM CONTROL
DEPTH VS TIME                                               4
                                   5.0        7.0
DRUREY SUBMARINE TRIM CONTROL
PITCH  VS TIME                                              4
                                   5.0        7.0
```

BIBLIOGRAPHY

1. Naval Ship Research and Development Center Report No. P-433-H-01, *Users Guide NSRDC Digital Program for Simulating Submarine Motion*, by Ronald W. Richard, June 1971.

2. *Principles of Naval Architecture*, Revised Edition, The Society of Naval Architects and Marine Engineers.

3. Naval Ship Research and Development Center Report 2510, *Standard Equations of Motion for Submarine Simulation*, by Morton Gertler and Grant R. Hagen, June 1967.

4. Romero, E., *Mathematical Models and Computers Solution for the Equations of Motion of Surface Ships and Submarines, in Six Degrees of Freedom*, Masters Thesis, Naval Postgraduate School, 1972.

5. *Fundamentals of Submarine Hydrodynamics, Motion, and Control*, Navships 0911-003-6010.

6. Kirk, D.E., *Optimal Control Theory*, Prentice-Hall Inc., 1970.

7. Thaler, G.J. and Huang, J., *Steady-State Decoupling and Design of Linear Multivariable Systems*, Research Report, University of Santa Clara, 1974.

INITIAL DISTRIBUTION

		No. Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0212 Naval Postgraduate School Monterey, California 93940	2
3.	Department Chairman, Code 52 Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940	2
4.	Professor G. Thaler, Code 52 Tr Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940	10
5.	A. Gerba, Code 52Gz Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940	1
6.	Professor D. Kirk, Code 52Ki Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940	1
7.	Naval Ship Research and Development Center Annapolis Laboratory Attn: Walt Blumberg Annapolis, Maryland 21402	1
8.	Department of the Navy Naval Ship Research and Development Center Code 1576 Attn: Bill Smith Bethesda, Maryland 20034	1
9.	Department of the Navy Naval Ship Research and Development Center Code 1576 Attn: John Ware Bethesda, Maryland 20034	1
10.	LT Harold L. Drurey, USN 39 Allen Place Hartford, Connecticut 06106	1